EasyPIC® 6

Manual de usuario

Los sistemas de desarrollo de MIkroElektronika son unas herramientas fiables y rápidas que pueden satisfacer las necesidades tanto de los ingenieros con experiencia como de los principiantes. Se caracterizan por un gran número de periféricos, ejemplos prácticos de códigos listos para ser utilizados y un conjunto amplio de placas adicionales.

Sistema de desarrollo

MikroElektronika

SOFTWARE AND HARDWARE SOLUTIONS FOR EMBEDDED WORLD ...making it simple

ESTIMADOS CLIENTES,
Querría darles las gracias por estar interesados en nuestros productos y por tener confianza en MikroElektronika.
Nuestro objetivo es proporcionarles con los productos de la mejor calidad. Además, seguimos mejorando nuestros rendimientos para responder a sus necesidades.

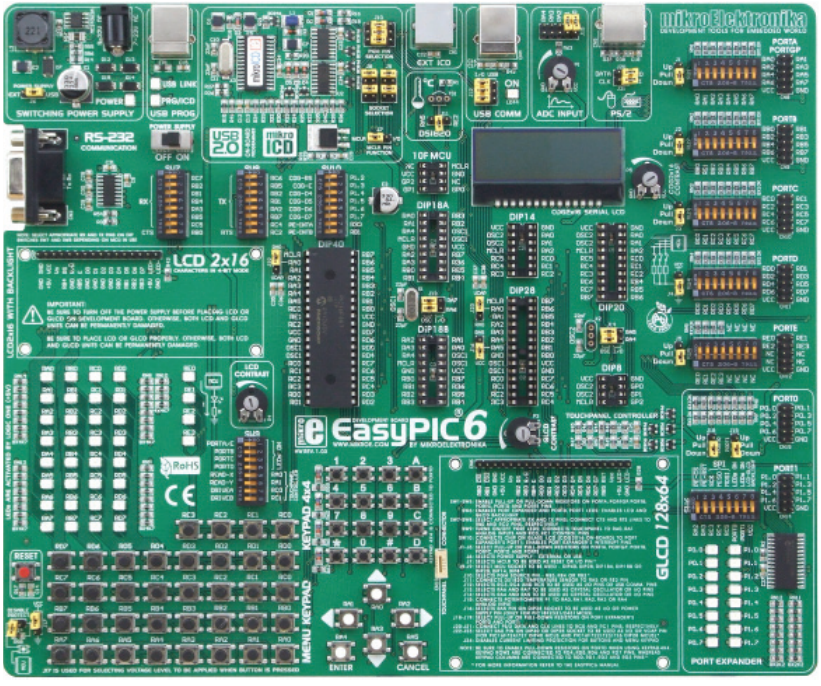
Nebojsa Matic
Director general

TABLA DE CONTENIDO

Introducción al sistema de desarrollo EasyPIC6 4
Prestaciones principales 5
1.0. Conexión del sistema al PC 6
2.0. Microcontroladores soportados 7
3.0. Programador incorporado 8
4.0. mikroICD (depurador en circuito a nivel de hardware) 10
5.0. Fuente de alimentación 11
6.0. Interfaz de comunicación RS-232 12
7.0. Interfaz de comunicación PS/2 13
8.0. Conector ICD 13
9.0. Comunicación USB 14
10.0. Sensor de temperatura DS1820 15
11.0. Conversor A/D 16
12.0. Diodos LED 17
13.0. Botones de presión 18
14.0. Teclados 19
15.0. Visualizador LCD 2x16 20
16.0. Visualizador LCD incorporado 2x16 21
17.0. Visualizador gráfico LCD 128x64 22
18.0. Panel táctil 23
19.0. Puertos de Entrada/Salida 24
20.0. Expansor de Puertos 26

Introducción al sistema de desarrollo EasyPIC® 6

El sistema de desarrollo *EasyPIC6* es una herramienta de desarrollo extraordinaria, adecuada para la programación y la experimentación con los microcontroladores PIC® de la compañía Microchip®. La placa dispone de un programador incorporado con el soporte de mikroICD® (depurador en circuito a nivel de hardware) que proporciona un interfaz entre el microcontrolador y el PC. Se espera de Ud. que escriba un código en alguno de nuestros compiladores, que genere un fichero .hex y que programe el microcontrolador utilizando el programador *PICflash®*. Los módulos numerosos, como visualizador gráfico LCD de 128x64 píxeles, visualizador LCD de 2x16 caracteres, visualizador LCD incorporado de 2x16 caracteres, teclado 4x4, expansor de puertos etc, le permiten simular con facilidad el funcionamiento del dispositivo destino.



PIC DEVELOPMENT BOARD — Placa de desarrollo completa y fácil de utilizar para los microcontroladores PIC

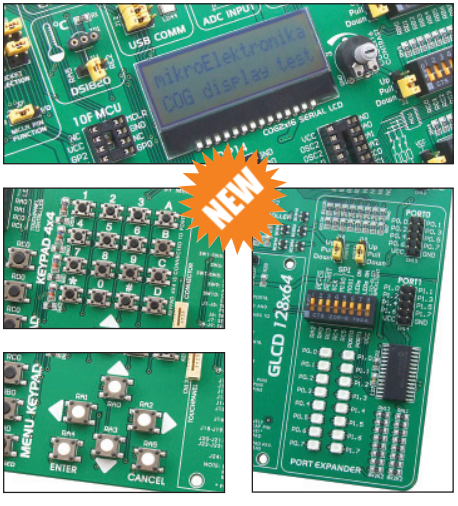
USB 2.0 IN-CIRCUIT PROGRAMMER — Programador incorporado de altas prestaciones con comunicación USB 2.0.

mikro ICD IN-CIRCUIT DEBUGGER — Depurador en circuito incorporado para la depuración paso a paso a nivel de hardware

PORT EXPANDER — Expansor de puertos incorporado proporciona expansión de E/S (2 puertos adicionales) utilizando interfaz en serie

COG 2x16 mikroE SERIAL LCD — Visualizador LCD incorporado de 2x16 caracteres con comunicación serie

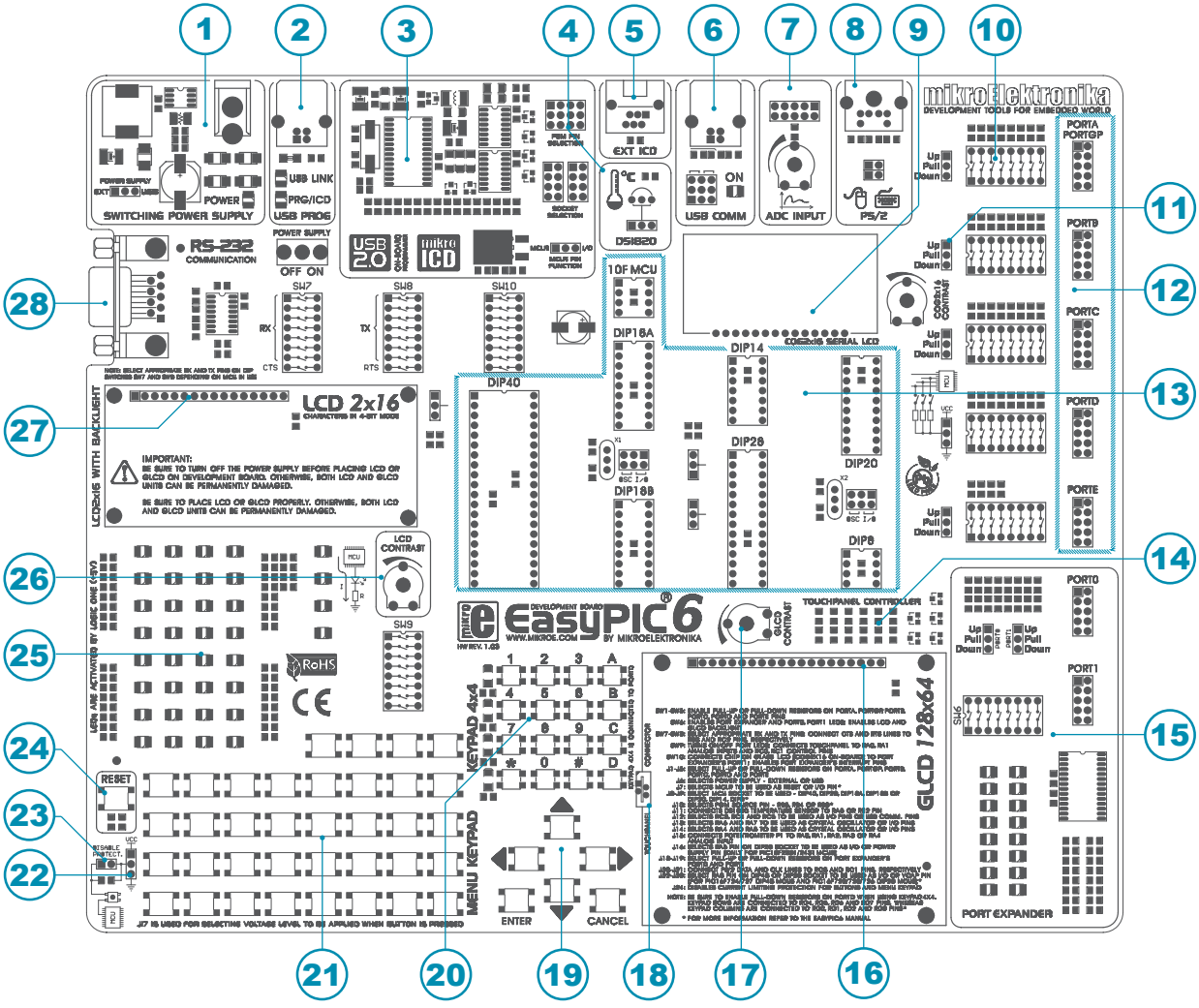
GRAPHIC LCD CONNECTOR ON-BOARD — Visualizador gráfico LCD con iluminación de fondo



El programa *PICflash* proporciona una lista completa de todos los microcontroladores soportados. La última versión de este programa con la lista actualizada de los microcontroladores se puede descargar desde nuestra página web: **www.mikroe.com**

El paquete contiene:	
Sistema de desarrollo:	*EasyPIC6*
CD:	CD del producto con el software apropiado
Cables:	cable USB
Documentación:	manuales *EasyPIC6, mikroICD, PICflash with mikroICD*, guía rápida *Instalación de los controladores USB* y *Esquema électrico* del sistema de desarrollo *EasyPIC6*

Especificación del sistema:	
Fuente de alimentación:	por el conector DC (7 -23V AC o 9-32V DC) o por el cable USB (5V DC)
Consumo de corriente:	40mA (depende del número de módulos incorporados en la placa actualmente activos)
Tamaño:	26,5 x 22cm (10,43 x 8,66inch)
Peso:	~417g (0.92lbs)



Prestaciones principales

1. Regulador del voltaje de la fuente de alimentación
2. Conector USB del programador incorporado
3. Programador USB 2.0 con el soporte de mikroICD
4. Zócalo del sensor de temperatura DS1820
5. Conector para el depurador externo (ICD2 o ICD3) de la compañía Microchip
6. Conector para la comunicación USB
7. Entradas para probar el conversor A/D
8. Conector PS/2
9. Visualizador LCD 2x16 incorporado
10. Interruptores DIP que permiten el funcionamiento de los resistores pull-up/pull-down
11. Elección del modo pull-up/pull-down
12. Conectores de los puertos de E/S
13. Zócalos para inserción de microcontroladores PIC
14. Controlador del panel táctil
15. Expansor de puertos (port expander)
16. Conector del visualizador gráfico LCD 128x64
17. Potenciómetro de contraste del visualizador gráfico LCD 128x64
18. Conector de panel táctil
19. Teclado *MENU*
20. Teclado 4x4
21. Botones de presión para simular las entradas digitales
22. Selector del estado lógico
23. Puente para encender/apagar el resistor protector
24. Botón para resetear el microcontrolador
25. 36 diodos LED que indican el estado lógico en los pines
26. Ajuste de contraste del visualizador LCD alfanumérico
27. Conector del visualizador LCD alfanumérico
28. Conector para comunicación RS-232

1.0. Conexión del sistema al PC

Paso 1:

La conexión del sistema *EasyPIC6* al PC se realiza por medio del cable USB. Una punta del cable USB proporcionado con el conector de tipo “USB B” se conecta al sistema de desarrollo como se muestra en la Figura 1.2. La otra punta del cable (USB A) se conecta al PC. Al establecer la conexión, asegúrese de que el puente J6 se coloque en la posición USB como se muestra en la Figura 1.1.

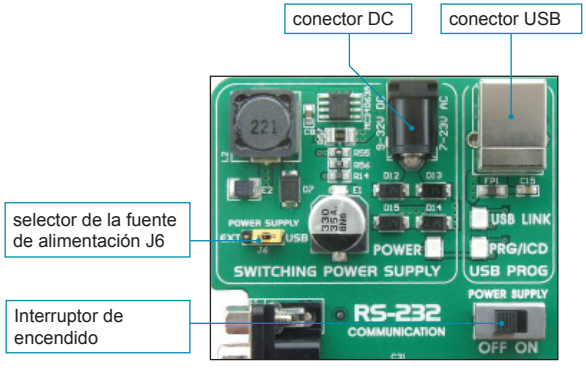


Figura 1.1. Fuente de Alimentación

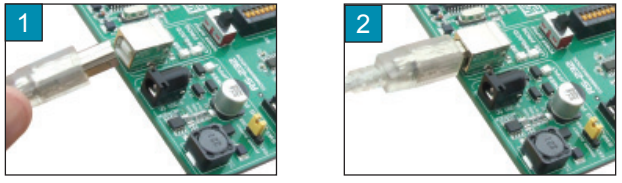


Figura 1.2. Conexión del cable USB (puente J6 en la posición USB)

Paso 2:

Siga las instrucciones para instalación de los controladores USB y del programador *PICflash* proporcionadas en los manuales relevantes. No es posible programar los microcontroladores sin haber instalado estos dispositivos anterirormente. En caso de que Ud. ya tenga algún compilador de MikroElektronika instalado en el PC, no es necesario reinstalar el programador *PICflash* ya que se instala automáticamente al instalar el compilador.

Paso 3:

Encienda el sistema de desarrollo al poner el interruptor de encendido en la posición ON. Se encenderán dos diodos LED etiquetados como “POWER” y “USB LINK”. El sistema está listo para su uso. Utilice el programador PICflash para introducir el código en el microcontrolador. Después de hacerlo, utilice la placa para probar y desarrollar sus proyectos.

NOTA: Si utiliza algunos módulos adicionales, tales como LCD, GLCD, placas adicionales etc, es necesario colocarlos apropiadamente en el sistema de desarrollo antes de que se encienda. De lo contrario, pueden quedarse dañados permanentemente.

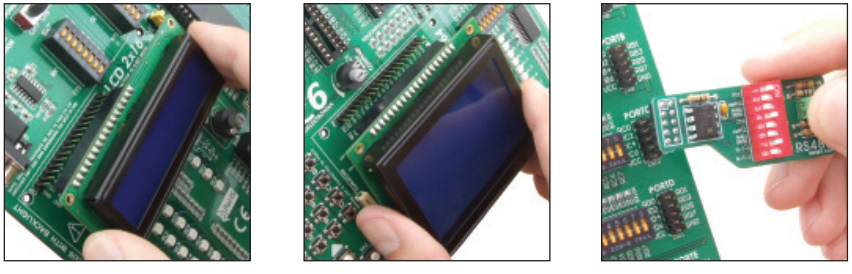


Figura 1.3. Colocación de los módulos adicionales en la placa

2.0. Microcontroladores soportados

El sistema de desarrollo *EasyPIC6* dispone de ocho zócalos separados para inserción de microcontroladores en los encapsulados DIP40, DIP28, DIP20, DIP18, DIP14 y DIP8. Estos zócalos permiten que los dispositivos soportados en los encapsulados DIP estén conectados directamente a la placa de desarrollo. Dos zócalos para microcontroladores están en el encapsulado DIP 18 en la placa. El zócalo a utilizar depende de la disposición de los pines en el microcontrolador utilizado. El sistema de desarrollo *EasyPIC6* viene con el microcontrolador en el encapsulado DIP 40.

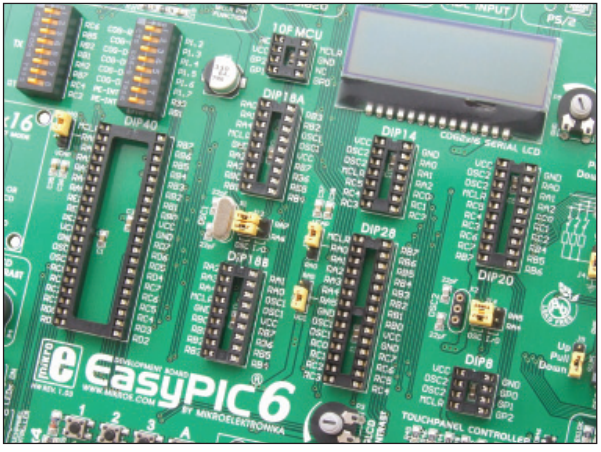


Figura 2.1 Zócalos del microcontrolador

Los puentes junto a los zócalos DIP40 y DIP28 tienen las siguientes funciones:

Puente	Posición
J22	RA0 - pin de E/S VCAP - capacitor de filtro (para 16F724/727)
J23	VCAP - capacitor de filtro (para 16F722/723) RA0 -pin de E/S
J16	RA5 - pin de E/S VCC - fuente de alimentación para microcontroladores PIC 18F2331/2431
J13	OSC - RA6, RA7 son pines de OSC. I/O - RA6, RA7 son pines de E/S
J14	OSC - RA4, RA5 son pines de OSC. I/O - RA4, RA5 son pines de E/S

Los microcontroladores PIC utilizan con frecuencia el cristal de cuarzo para estabilizar la frecuencia de reloj. El *EasyPIC6* dispone de dos zócalos para colocar el cristal de cuarzo. Los microcontroladores en los encapsulados DIP18A, DIP18B, DIP28 y DIP40 utilizan el zócalo (OSC1) X1 para el cristal de cuarzo. En caso de que se utilicen los microcontroladores en los encapsulados DIP8, DIP14 y DIP20 es necesario mover el cristal de cuarzo del zócalo X1 al zócalo X2 (OSC2). Además, es posible reemplezar el cristal de cuarzo actual por otro. El valor del cristal de cuarzo depende de la frecuencia de reloj máxima permitida. Los microcontroladores colocados en el zócalo 10F utilizan un oscilador interno. No están conectados a ningún zócalo del cristal de cuarzo anteriormente mencionado.

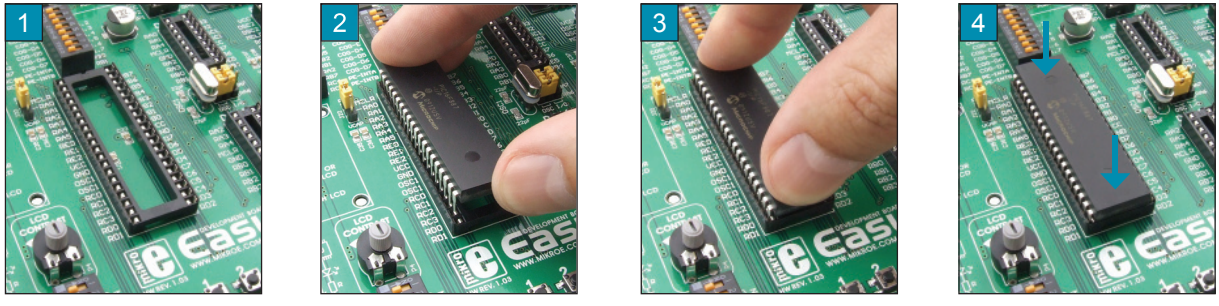


Figura 2.2 Inserción del microcontrolador en el zócalo apropiado

Antes de colocar el microcontrolador en el zócalo adecuado, asegúrese de que la fuente de alimentación esté apagada. En la Figura 2.2. se muestra cómo colocar correctamente un microcontrolador. La Figura 1 muestra el zócalo DIP 40 vacío. Coloque una punta del microcontrolador en el zócalo como se muestra en la Figura 2. Entonces ponga lentamente el microcontrolador más abajo hasta que los pines encajen en el zócalo, como se muestra en la Figura 3. Compruebe una vez más si todo está colocado correctamente y presione el microcontrolador lentamente hasta que encaje en el zócalo completamente, como se muestra en la Figura 4.

NOTA: En la placa del sistema de desarrollo se puede colocar un sólo microcontrolador

3.0. El programador incorporado *USB 2.0 PICflash with mikroICD*

El programador *PICflash* es una herramienta obligatoria al trabajar con microcontroladores. El sistema de desarrollo *EasyPIC6* dispone del programador *PICflash* integrado con el soporte de *mikroICD*. El programador *PICflash* permite establecer una conexión entre el PC y el microcontrolador. Utilice el programador *PICflash* para cargar un fichero HEX en el microcontrolador. En la Figura 3.2. se muestra la conexión entre el compilador, el programador *PICflash* y el microcontrolador.

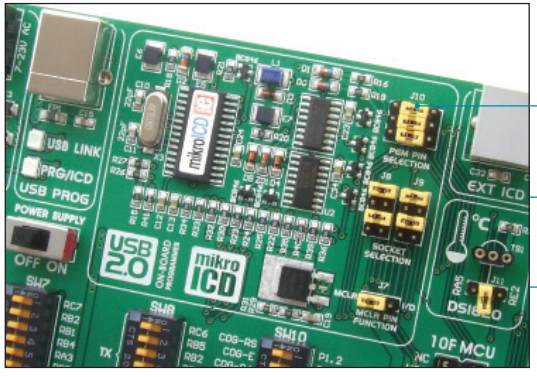

Grupo de puentes J10 utilizado para conectar la línea PGM
Grupo de puentes J8 y J9 utilizado para seleccionar el zócalo en el que se encuentra el microcontrolador
Grupo de puentes J7 utilizado para selección de la función del pin MCLR

Figura 3.1. Programador *PICflash* con *mikroICD*

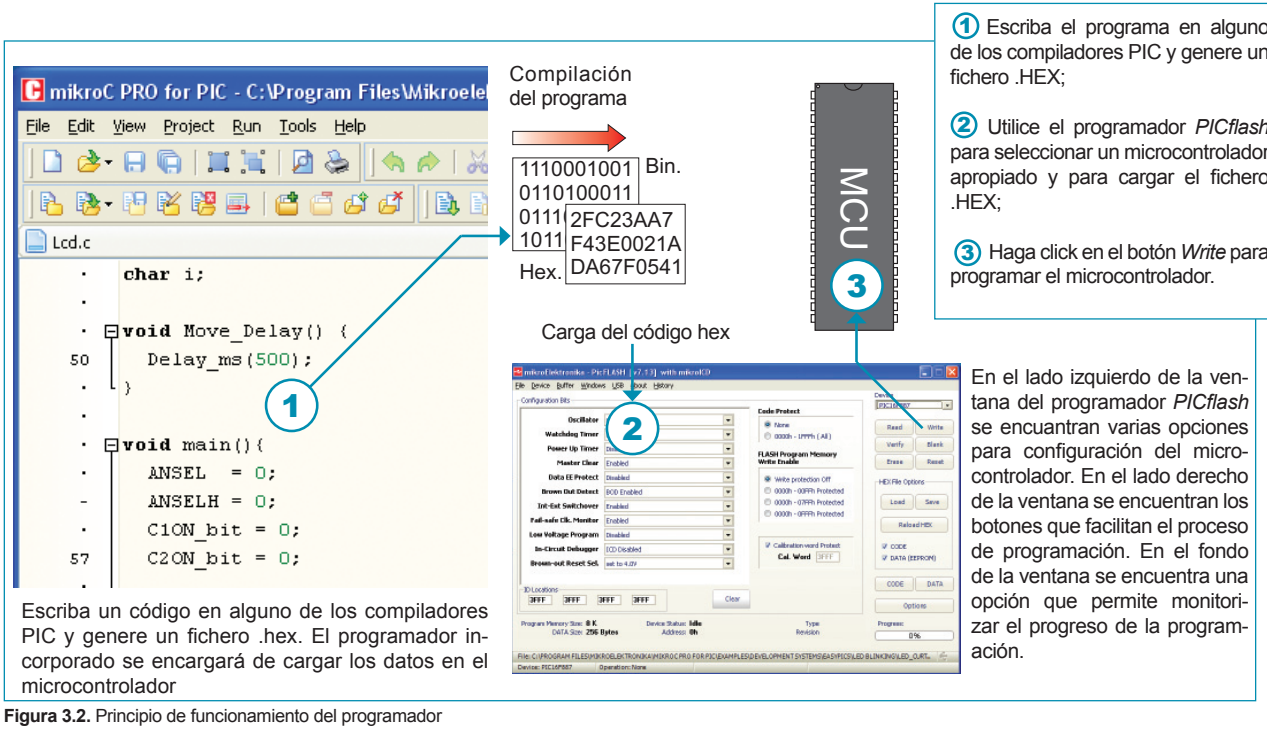

mikroC PRO for PIC - C:\Program Files\Mikroele
File Edit View Project Run Tools Help
Lcd.c
char i;
void Move_Delay() {
Delay_ms(500);
}
void main(){
ANSEL = 0;
ANSELH = 0;
C1ON_bit = 0;
C2ON_bit = 0;
1
Escriba un código en alguno de los compiladores PIC y genere un fichero .hex. El programador incorporado se encargará de cargar los datos en el microcontrolador
Compilación del programa
1110001001
0110100011
0111
1011
Bin.
2FC23AA7
F43E0021A
DA67F0541
Hex.
MCU
3
Carga del código hex
2
1 Escriba el programa en alguno de los compiladores PIC y genere un fichero .HEX;
2 Utilice el programador *PICflash* para seleccionar un microcontrolador apropiado y para cargar el fichero .HEX;
3 Haga click en el botón *Write* para programar el microcontrolador.
En el lado izquierdo de la ventana del programador *PICflash* se encuantran varias opciones para configuración del microcontrolador. En el lado derecho de la ventana se encuentran los botones que facilitan el proceso de programación. En el fondo de la ventana se encuentra una opción que permite monitorizar el progreso de la programación.

Figura 3.2. Principio de funcionamiento del programador

NOTA: Para obtener más informaciones del programador *PICflash* consulte el manual relevante proporcionado con el paquete del sistema de desarrollo *EasyPIC6*.

Los microcontroladores PIC se pueden programar de dos maneras: los modos de programación de alto y bajo voltaje. El programador *PIC-flash* utiliza únicamente el modo de programación de alto voltaje. Este modo requiere llevar un voltaje más alto que el voltaje de la fuente de alimentación del microcontrolador (el rango de 8 a 14V, lo que depende de tipo del microcontrolador utilizado) al pin MCLR/pp para realizar la programación.
El modo de programación de bajo voltaje se puede habilitar/deshabilitar al utilizar los bits de configuración del microcontrolador. Si el modo de programación de bajo voltaje está habilitado, el proceso de programación se inicia al aplicar un uno lógico (1) al pin PGM. A diferencia de este modo, el de alto voltaje está siempre habilitado y el proceso de programación se inicia al aplicar un voltaje alto al pin MCLR/VPP.
Todos los microconroladores PIC disponen del modo de programación de bajo voltaje por defecto. En pocos casos, para programar el microcontrolador en modo de programación de alto voltaje es necesario aplicar un cero lógico (0) al pin PGM, lo que deshabilita que el microcontrolador entre en modo de programación de bajo voltaje. Dependiendo del microcontrolador utilizado es posible seleccionar uno de los siguientes pines RB3, RB4 y RB5 como el pin PGM. El grupo de puentes J10 se utiliza para seleccionar el pin PGM como se muestra en la Figura 3.3.

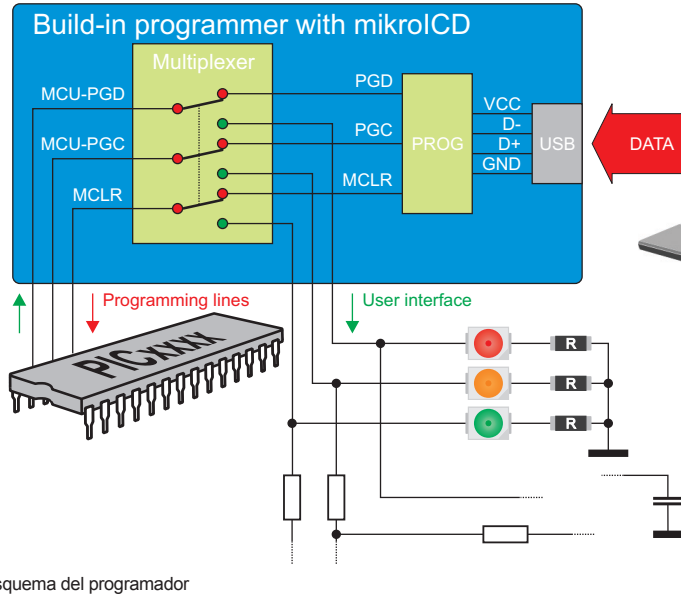
La posición del puente J10 en la que los pines RB3, RB4 y RB5 no están conectados a la línea PGM.

Posición del puente J10 en la que la línea PGM está conectada al pin RB5.

Posición del puente J10 en la que la línea PGM está conectada al pin RB4.

Posición del puente J10 en la que la línea PGM está conectada con el pin RB3.

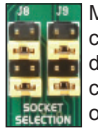
Figura 3.3. Diversas posiciones del puente J10



Durtante la programación, un multiplexor desconecta los pines del microcontrolador, utilizados para la programación, del resto de la placa y los conecta al programador *PIC-flash*. Una vez terminada la programación, los pines se desconectan del programador y se pueden utilizar como los pines de E/S.

Figura 3.4. Esquema del programador

Microcontroladores colocados en uno de los siguientes zócalos: DIP40, DIP28 DIP18A o DIP18B. (Posición por defecto).



Microcontroladores colocados en uno de los siguientes zócalos: DIP20, DIP14 o DIP8.

Figura 3.5. Posición de los puentes J8 y J9

Los puentes J8 y J9 se utilizan para seleccionar el zócalo que recibe una señal de programación. En la Figura 3.5. se muestran la posiciones de los puentes J8 y J9 que dependen de los zócalos DIP utilizados.

MCLR utilizado como el pin MCLR/Vpp

MCLR utilizado como pin de I/O

Figura 3.6. Posición del puente J7

La función del pin MCLR (*Master Clear*, literalmente borrado maestro) depende de la posición del puente J7. Cuando se coloca en la posición a la izquierda el pin MCRL tiene una función por defecto, es decir se utiliza como MCRL /Vpp. Cuando se coloca en la posición a la derecha el pin MCRL está disponible como un pin de E/S.

4.0. mikroICD (Depurador en circuito a nivel de hardware)

mikroICD (depurador en circuito) es una parte inetgral del programador incorporado. Se utiliza con el próposito de probar y depurar programas en tiempo real. El proceso de probar y depurar se realiza al monitorizar los estados de todos los registros dentro del microcontrolador durante su funcionamiento en entorno real. El software *mikroICD* está incorporado en todos los compiladores PIC diseñados por MikroElektronika (mikroBASIC®, mikroC® and mikroPASCAL®). Tan pronto como se inicie el depurador *mikroICD*, aparecerá una ventana, como se muestra en la figura más abajo.

El depurador *mikroICD* se comunica con el PC por los pines de programación que por eso no se pueden utilizar como pines de E/S durante la ejecución de la depuración de programa.

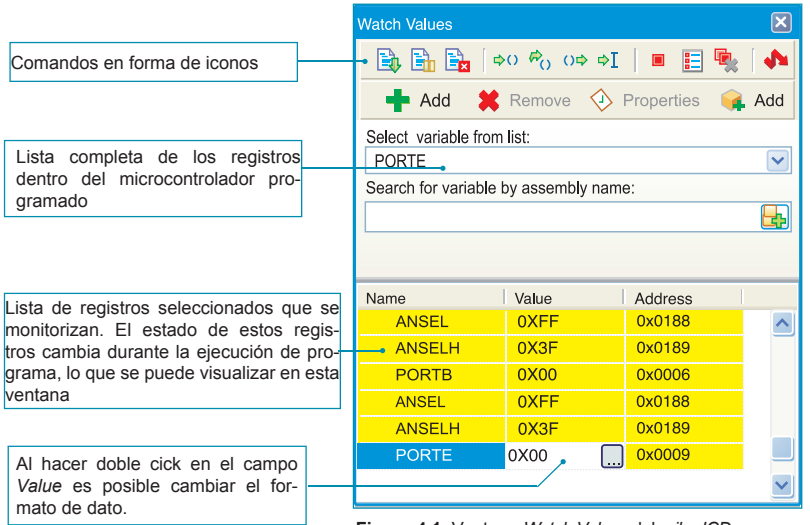

Comandos en forma de iconos
Lista completa de los registros dentro del microcontrolador programado
Lista de registros seleccionados que se monitorizan. El estado de estos registros cambia durante la ejecución de programa, lo que se puede visualizar en esta ventana
Al hacer doble cick en el campo *Value* es posible cambiar el formato de dato.
Watch Values
Add
Remove
Properties
Add
Select variable from list:
PORTE
Search for variable by assembly name:
Name
Value
Address
ANSEL 0XFF 0x0188
ANSELH 0X3F 0x0189
PORTB 0X00 0x0006
ANSEL 0XFF 0x0188
ANSELH 0X3F 0x0189
PORTE 0X00 0x0009

Figura 4.1: Ventana *Watch Values* del *mikroICD*

Opciones del depurador *mikroICD*:

Start Debugger	[F9]
Run/Pause Debugger	[F6]
Stop Debugger	[Ctrl + F2]
Step Into	[F7]
Step Over	[F8]
Step Out	[Ctrl + F8]
Toggle Breakpoint	[F5]
Show/Hide Breakpoints	[Shift + F4]
Clear Breakpoints	[Ctrl + Shift + F4]

Cada opción se activa por medio de los atajos de teclado o al hacer click en el icono apropiado en la ventana *Watch Value*.

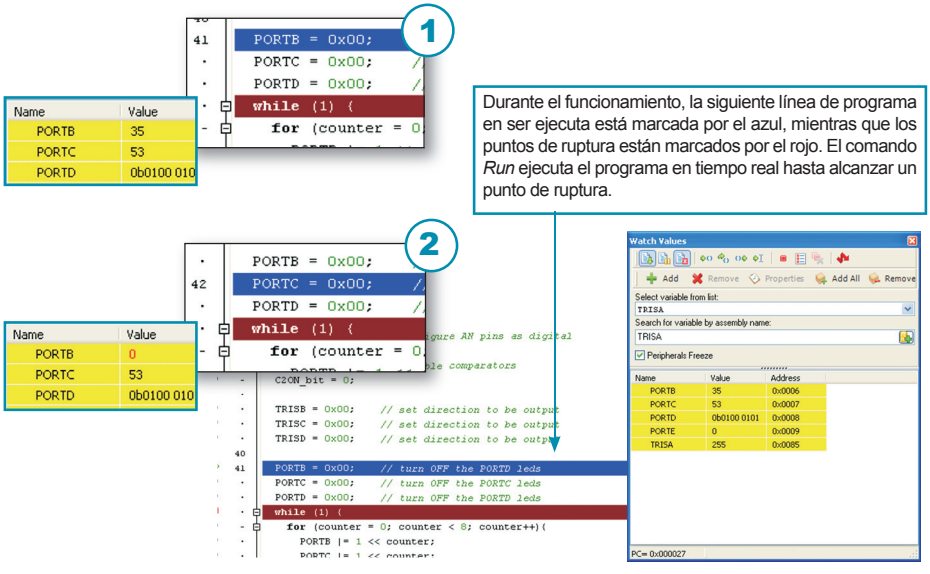
El depurador *mikroICD* también ofrece funciones tales como ejecutar el programa paso a paso, detener la ejecución de programa para monitorizar el estado de los registros actualmente activos por medio de los puntos de ruptura, monitorizar los valores de algunas variables etc. El siguiente ejemplo muestra una ejecución de programa paso a paso utilizando el comando *Step Over*.

Paso 1:

En este ejemplo la línea de programa 41 está marcada por el azul, lo que quiere decir que es la siguiente en ser ejecutada. El estado actual de todos los registros dentro del microcontrolador se puede visualizar en la ventana *Watch Values* del *mikroICD*.

Paso 2:

Después de ejecutar el comando *Step Over* el microcontrolador ejecutará la línea de programa 41. La siguiente línea en ser ejecutada está marcada por el azul. El estado de los registros cambiados durante la ejecución de esta instrucción se puede visualizar en la ventana *Watch Values*.


1
PORTB = 0x00;
PORTC = 0x00;
PORTD = 0x00;
while (1) {
for (counter = 0
Name Value
PORTB 35
PORTC 53
PORTD 0b0100 010
Durante el funcionamiento, la siguiente línea de programa en ser ejecuta está marcada por el azul, mientras que los puntos de ruptura están marcados por el rojo. El comando *Run* ejecuta el programa en tiempo real hasta alcanzar un punto de ruptura.
2
PORTB = 0x00;
PORTC = 0x00;
PORTD = 0x00;
while (1) {
for (counter = 0
Name Value
PORTB 0
PORTC 53
PORTD 0b0100 010
TRISB = 0x00; // set direction to be output
TRISC = 0x00; // set direction to be output
TRISD = 0x00; // set direction to be output
PORTB = 0x00; // turn OFF the PORTD leds
PORTC = 0x00; // turn OFF the PORTC leds
PORTD = 0x00; // turn OFF the PORTD leds
while (1) {
for (counter = 0; counter < 8; counter++){
PORTB |= 1 << counter;
Watch Values
Add
Remove
Properties
Add All
Remove
Select variable from list:
TRISA
Search for variable by assembly name:
TRISA
Peripherals Freeze
Name Value Address
PORTB 35 0x0006
PORTC 53 0x0007
PORTD 0b0100 0101 0x0008
PORTE 0 0x0009
TRISA 255 0x0085
PC= 0x000027

NOTA: Para obtener más informaciones del depurador *mikroICD* consulte el manual *mikroICD debugger*.

5.0. Fuente de alimentación

El sistema de desarrollo *EasyPIC6* puede utilizar uno de dos fuentes de alimentación:

1. Fuente de alimentación de +5V desde el PC por el cable USB de programación;
2. La fuente de alimentación externa conectada a un conector DC proporcionado en el sistema de desarrollo.

El regulador del voltaje de la fuente de alimentación MC34063A y el rectificador Gretz permiten que el voltaje de la fuente de alimentación externa sea AC (en el rango de 7V a 23V) o DC (en el rango de 9V a 32V). El puente J6 se utiliza como selector de la fuente de alimentación. Cuando se utilice la fuente de alimentación USB, el puente J6 debe estar en la posición USB. Cuando se utilice la fuente de alimentación externa, el puente J6 debe estar en la posición EXT. El sistema de desarrollo se enciende/apaga al cambiar de posición del interruptor de encendido/apagado respectivamente.

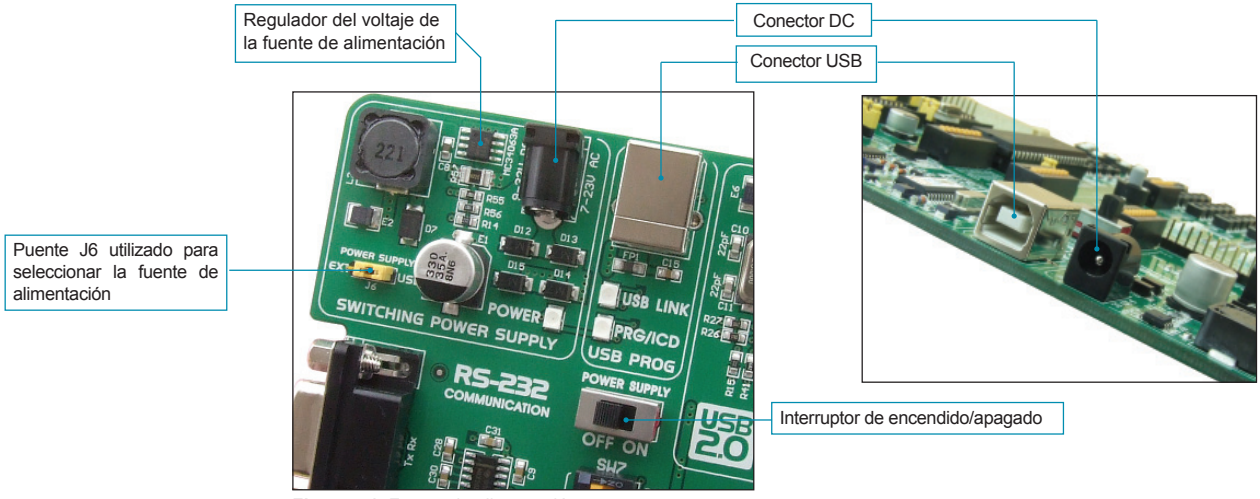


Figura 5.1. Fuente de alimentación

El programador utiliza el interruptor MOSFET para apagar la fuente de alimentación en el sistema de desarrollo durante la programación. Al acabar el proceso de la programación, el programador enciende la fuente de alimentación de nuevo.

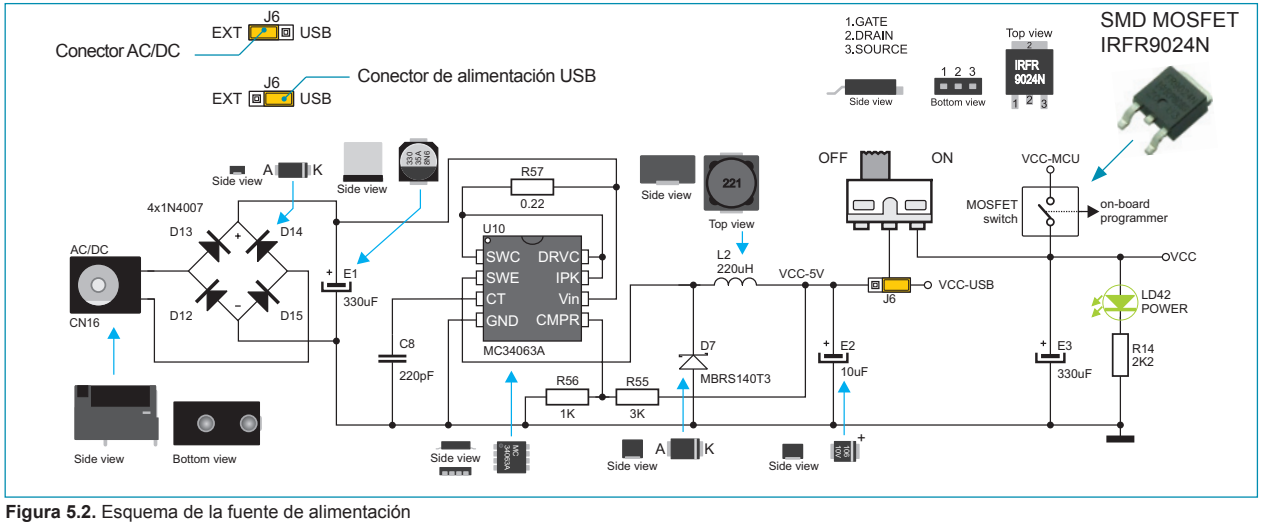


Figura 5.2. Esquema de la fuente de alimentación

6.0. Interfaz de comunicación RS-232

La comunicación serie **RS-232** se realiza por un conector SUB-D de 9 pines y el módulo USART del microcontrolador. Para habilitar tal comunicación, es necesario establecer una conexión entre las líneas de comunicación RX y TX y los pines del microcontrolador sobre los que está el módulo USART, utilizando un interruptor DIP (líneas de conformidad de conexión (*handshaking*) CTS y RTS se utilizan opcionalmente). Los pines del microcontrolador utilizados en tal comunicación serie están marcados de la siguiente manera: RX (*receive data) - recibir datos*; TX (*transmit data) - transmitir datos*; CTS (*clear to send*) - permitido para transmitir; RTS (*request to send)* - listo para enviar. La velocidad de transmisión en baudios es hasta 115 kbps.
USART (*Universal Synchronous/asynchronous Receiver/Transmitter*) - Receptor/Transmisor Sincrónico/Asincrónico Universal es uno de los tipos básicos de la comunicación serie entre el PC y las unidades periféricas. Para habilitar que el módulo de microcontrolador reciba las señales de entrada con diferentes niveles de voltaje, es necesario proporcionar un convertidor de nivel de voltaje como MAX-202C.

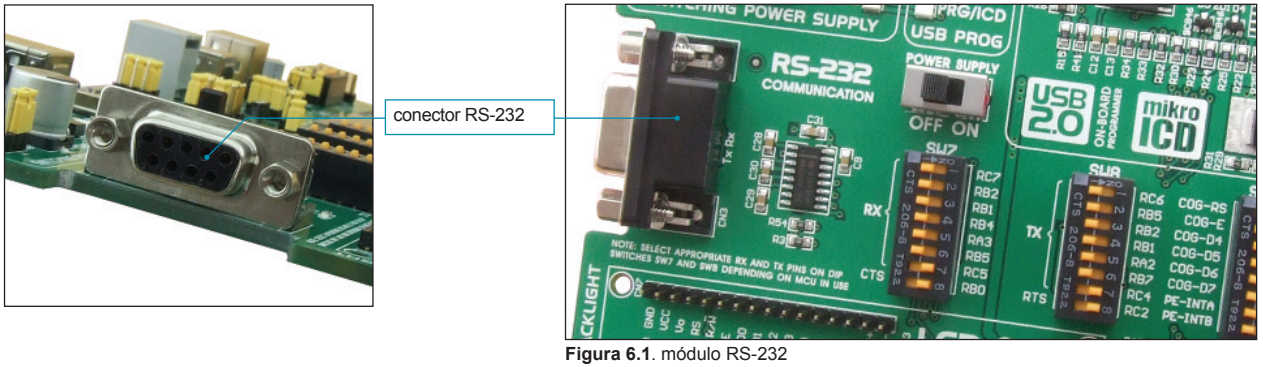


Figura 6.1. módulo RS-232

La función de los interruptores DIP SW7 y SW8 es de determinar cuál de los pines de microcontrolador se utilizará como líneas RX y TX. La disposición de los pines difiere dependiendo del tipo de microcontrolador. La Figura 6-2 muestra el microcontrolador en el encapsulado DIP40 (PIC16F887).

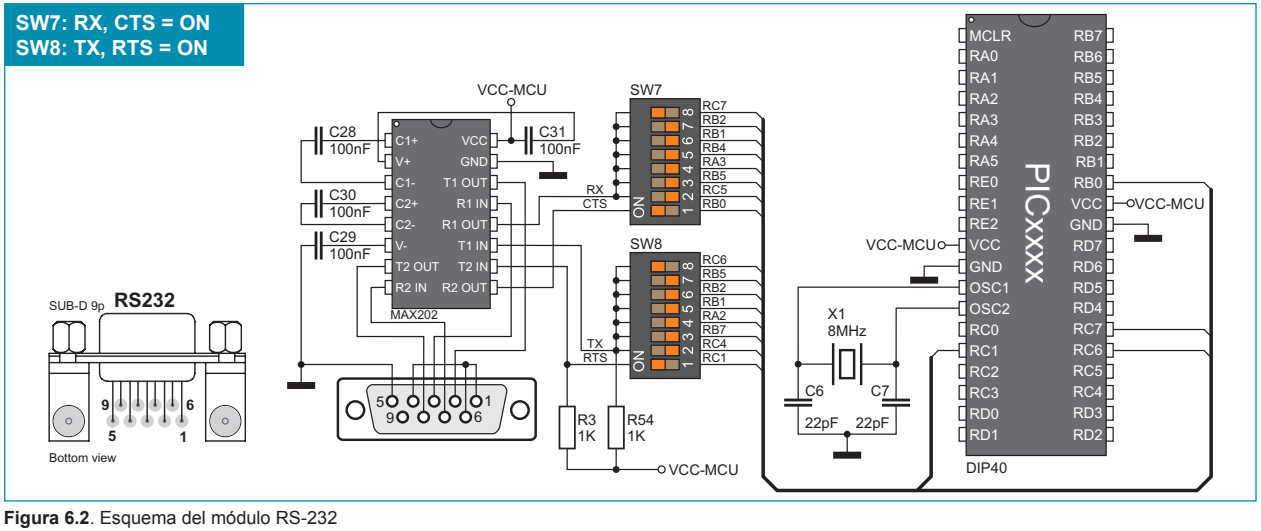


Figura 6.2. Esquema del módulo RS-232

NOTA: Asegúrese de que su microcontrolador esté proporcionado con el módulo USART, ya que no está integrado necesariamente en todos los microcontroladores.

7.0. Interfaz de comunicación PS/2

El conector **PS/2** permite conectar los dispositivos de entrada, tales como teclado y ratón, con el sistema de desarrollo. Para habilitar la comunicación PS/2 es necesario colocar apropiadamente los puentes J20 y J21, así que se las líneas DATA y CLK están conectadas a los pines del microcontrolador RC0 y RC1.

No conecte/desconecte las unidades de entrada al conector PS/2 mientras que el sistema de desarrollo está encendido debido a que puede dañar el microcontrolador permanentemente.

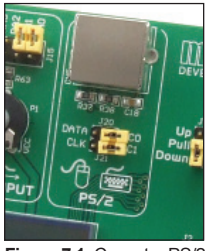


Figura 7.1. Conector PS/2 (J20 y J21 no están conectados)

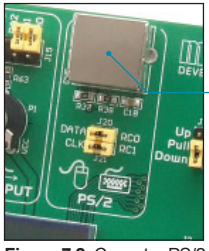
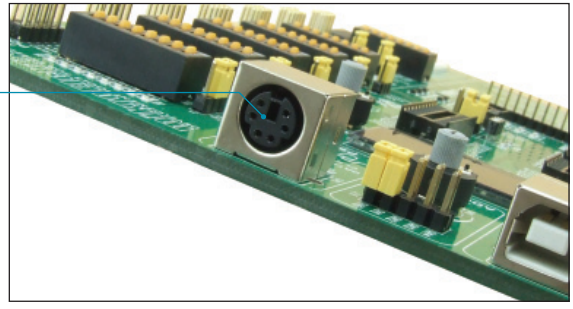


Figura 7.2. Conector PS/2 (J20 y J21 están conectados)



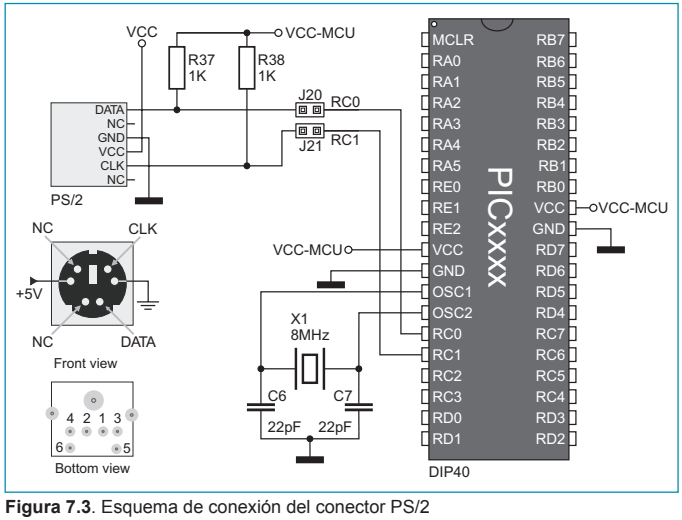


Figura 7.3. Esquema de conexión del conector PS/2

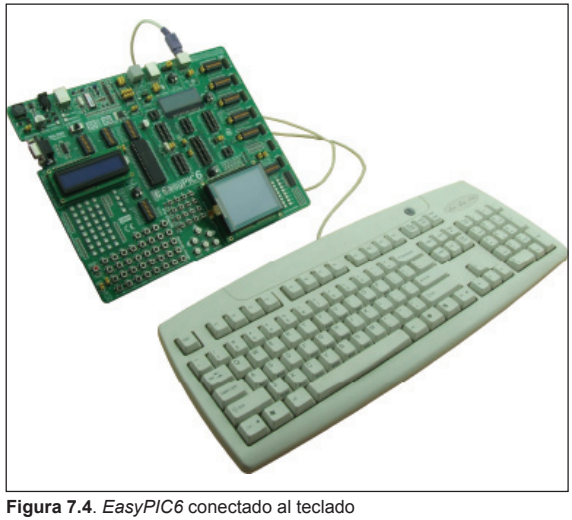


Figura 7.4. *EasyPIC6* conectado al teclado

8.0. Conector ICD

Este conector permite que el microcontrolador se comunique con el depurador ICD externo (ICD2 o ICD3)* de la compañía Microchip. Los puentes J8 y J9 se colocan de la misma manera como al utilizar el programador *PICflash* con el *mikroICD* diseñado por MikroElektronika.

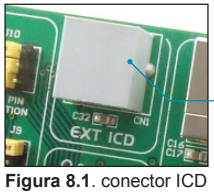
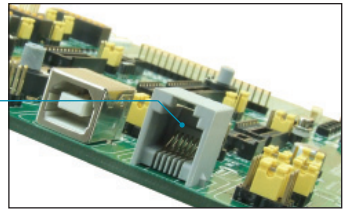


Figura 8.1. conector ICD



*ICD2 i ICD3 son las marcas comerciales registradas de la compañía Microchip

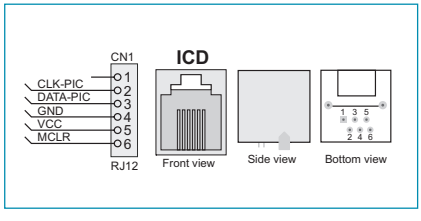


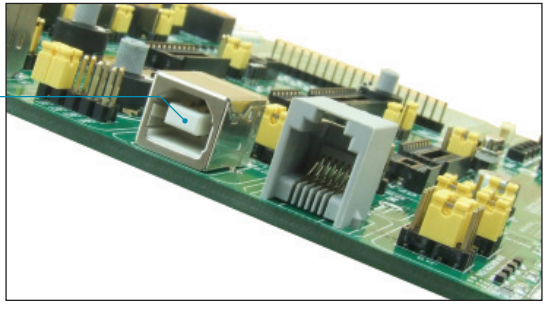
Figura 8.2. Esquema y etiquetas de los pines en el conector ICD

9.0. Comunicación USB

El conector **USB** permite conectar los microcontroladores PIC que disponen de un módulo de comunicación USB incorporado a los dispositivos periféricos. Para establecer la comunicación USB, es necesario cambiar de posición del puente J12 de izquierda a derecha. De este modo se conectan los cables USB DATA (D+ y D-) a los pines del microcontrolador RC4 y RC5, mientras que el pin RC3/VUSB se conecta a los capacitores C16 y C17. En caso de que no se utilice la comunicación USB, los puentes J12 deben quedarse en la posición a la izquierda. El estado de la comunicación USB (encendido/apagado) está indicado por el LED.
En las Figuras 9.3 y 9.4 se muestra el esquema de los microcontroladores utilizados con más frecuencia con el módulo USB integrado.

Figura 9.1. Comunicación USB deshabilitada (posición por defecto)

Figura 9.2. Comunicación USB habilitada



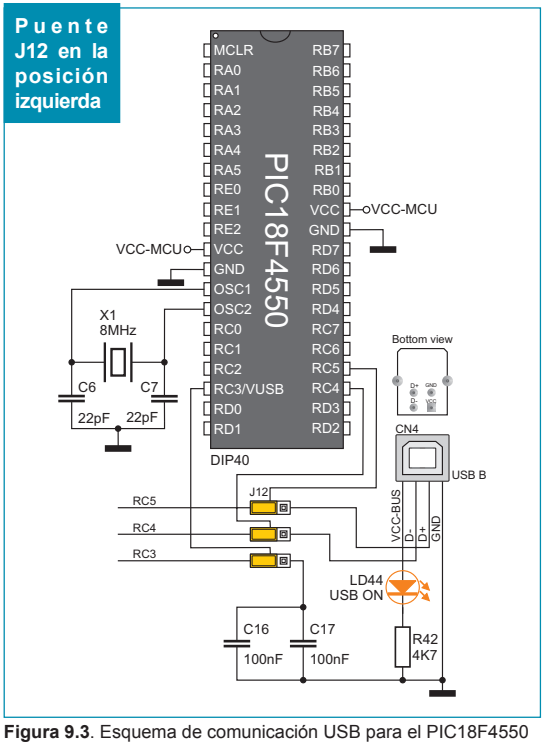


Figura 9.3. Esquema de comunicación USB para el PIC18F4550

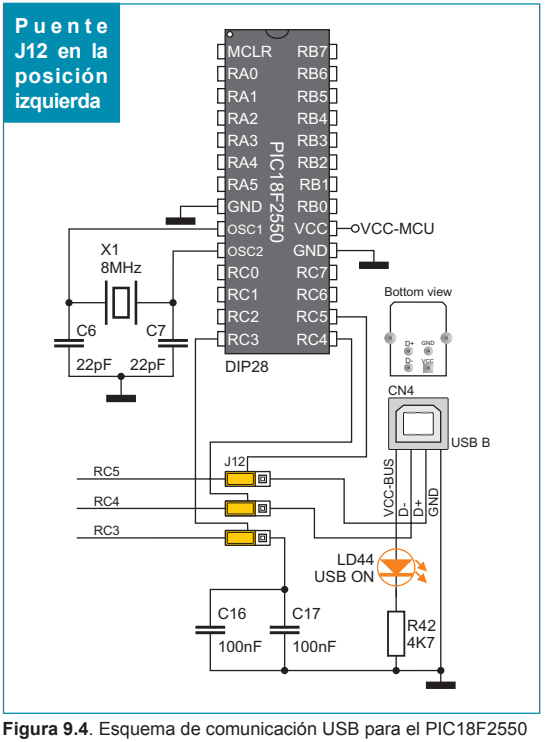


Figura 9.4. Esquema de comunicación USB para el PIC18F2550

10.0. Sensor de temperatura DS1820

La comunicación serie 1-wire® permite la transmisión de datos por medio de una sóla línea de comunicación, mientras que el proceso mismo está bajo el control del microcontrolador maestro. La ventaja de tal comunicación es que se utiliza sólo un pin de microcontrolador. Todos los dispositivos esclavos disponen de un código ID único por defecto, lo que permite que el dispositivo maestro identifique fácilmente los dispositivos que comparten el mismo interfaz.
DS1820 es un sensor de temperatura que utiliza el estándar 1-wire para su funcionamiento. Es capaz de medir las temperaturas dentro del rango de -55 a 125°C y proporcionar la exactitud de medición de ±0.5°C para las temperaturas dentro del rango de -10 a 85°C. Para su funcionamiento el DS1820 requiere un voltaje de la fuente de alimentación de 3 a 5.5V. El DS1820 tarda como máximo 750 ms en calcular la temperatura con una resolución de 9 bits. El sistema de desarrollo *EasyPIC6* proporciona un zócalo separado para el DS1820. Se puede utilizar uno de los pines, RA5 o RE2 para la comunicación con el microcontrolador. El próposito del puente J11 es de seleccionar el pin que será utilizado para la comunicación 1-wire. En la Figura 10.5. se muestra la comunicación 1-wire con el microcontrolador por el pin RA5.

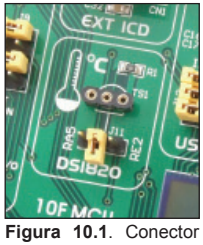


Figura 10.1. Conector DS1820 (no se utiliza la comunicación 1-wire)

Figura 10.2. Puente J11 en la posición a la izquierda (comunicación 1-wire por el pin RA5)

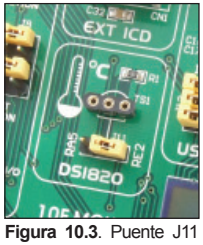


Figura 10.3. Puente J11 en la posición a la derecha (comunicación 1-wire por el pin RE2)

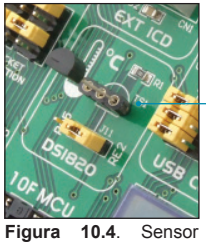


Figura 10.4. Sensor DS1820 colocado en el zócalo apropiado.

NOTA: Asegúrese de que el semicírculo en la placa coincida con la parte redonda del DS1820.

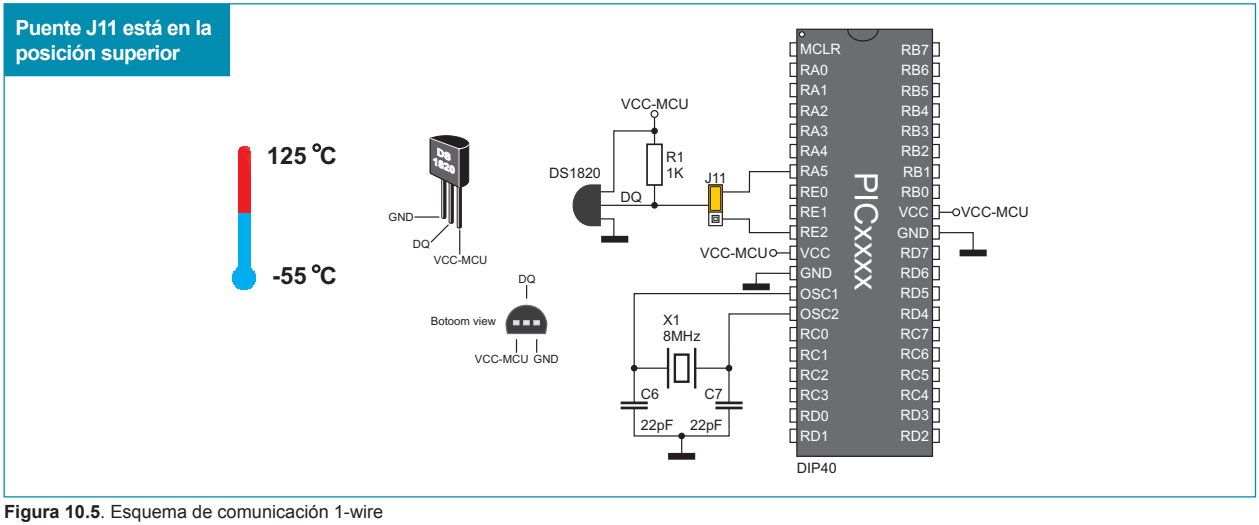


Figura 10.5. Esquema de comunicación 1-wire

11.0. Conversor A/D

Un conversor A/D (analógico-digital) se utiliza con el propósito de covertir una señal analógica a un valor digital apropiado. El conversor A/D es lineal, lo que quiere decir que el número convertido es linealmente dependiente del valor del voltaje de entrada.
El conversor A/D dentro del microcontrolador convierte un valor de voltaje analógico a un número de 10 bits. Los voltajes que varían de 0 a 5V DC se pueden suministrar por las entradas de prueba del conversor A/D.
El puente J15 se utiliza para seleccionar uno de los siguientes pines para la conversión AD: RA0, RA1, RA2, RA3 o RA4. El resistor R63 tiene la función de protección. Se utiliza para limitar el flujo de corriente por el potenciómetro o por el pin del microcontrolador. El valor del voltaje analógico de entrada se puede cambiar linealmente utilizando el potenciómetro P1.

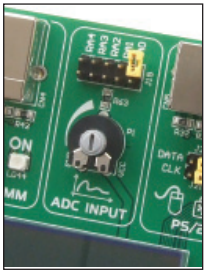


Figura 11.1. ADC (posición por defecto del puente)

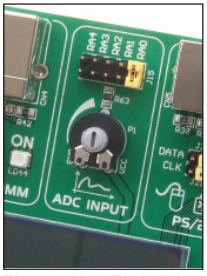


Figura 11.2. El pin RA0 utilizado como el pin de entrada para la conversión A/D

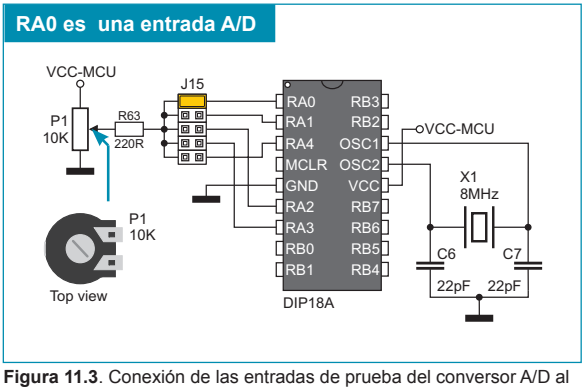


Figura 11.3. Conexión de las entradas de prueba del conversor A/D al microcontrolador PIC en el encapsulado DIP18A

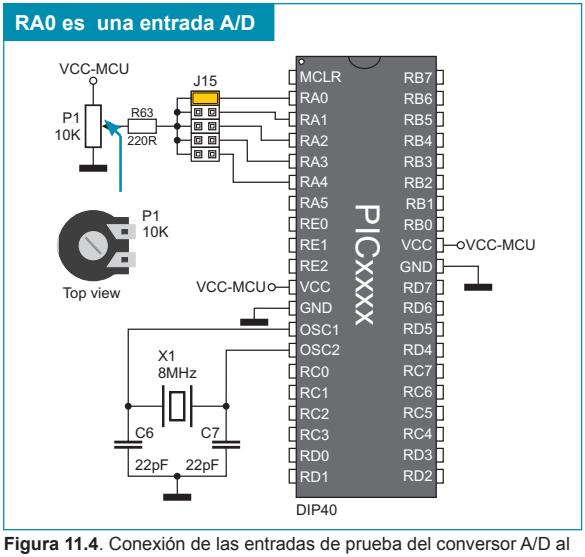


Figura 11.4. Conexión de las entradas de prueba del conversor A/D al microcontrolador PIC en el encapsulado DIP40

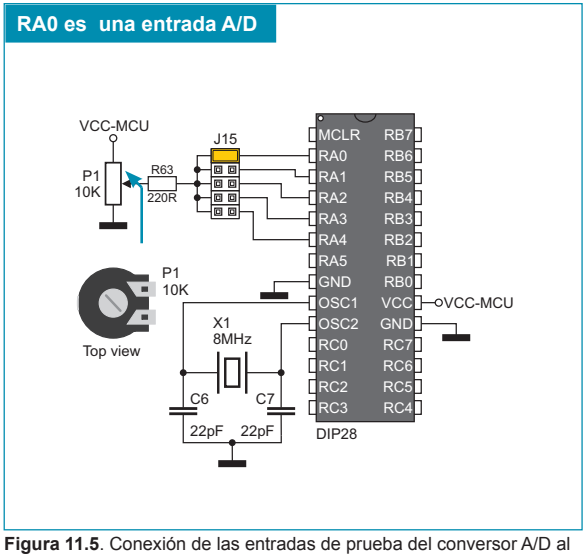


Figura 11.5. Conexión de las entradas de prueba del conversor A/D al microcontrolador PIC en el encapsulado DIP28

NOTA: Para que un microcontrolador pueda realizar una conversión A/D con exactitud, es necesario apagar los diodos LED y los resistores pull-up/pull-down en los pines de los puertos utilizados por el conversor A/D.

12.0. Diodos LED

El diodo LED (Light-Emitting Diode) - Diodo emisor de luz, representa una fuente electrónica de luz de muy alta eficacia. Al conectar los LEDs es necesario colocar el resistor para limitar la corriente cuyo valor se calcula utilizando la fórmula R=U/I donde R se refierre a la resistencia expresada en ohmios, U se refierre al voltaje en el LED, I se refierre a la corriente del LED. El voltaje común del LED es aproximadamente 2.5V, mientras que la intensidad de corriente varía de 1 a 20mA dependiendo del tipo del diodo LED. El sistema de desarrollo *EasyPIC6* utiliza los LEDs con la corriente I=1mA).
El sistema de desarrollo *EasyPIC6* tiene 36 LEDs que indican visualmente el estado lógico en cada pin de E/S del microcontrolador. Un diodo LED activo indica la presencia de un uno lógico (1) en el pin. Para habilitar que se muestre el estado de los pines, es necesario seleccionar el puerto apropiado PORTA/E, PORTB, PORTC o PORTD utilizando el interruptor DIP SW9.

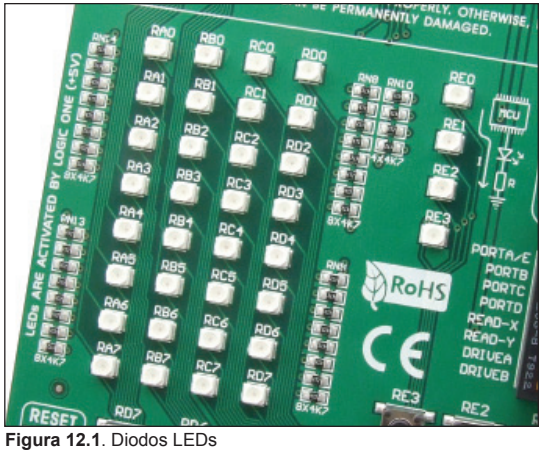
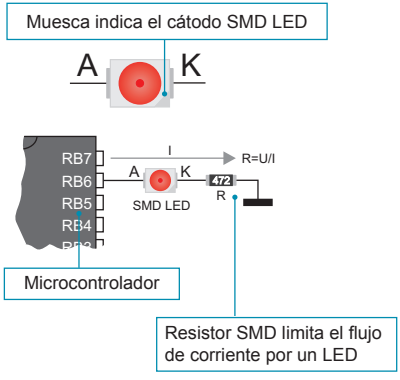


Figura 12.1. Diodos LEDs



SW9: PORTB = ON

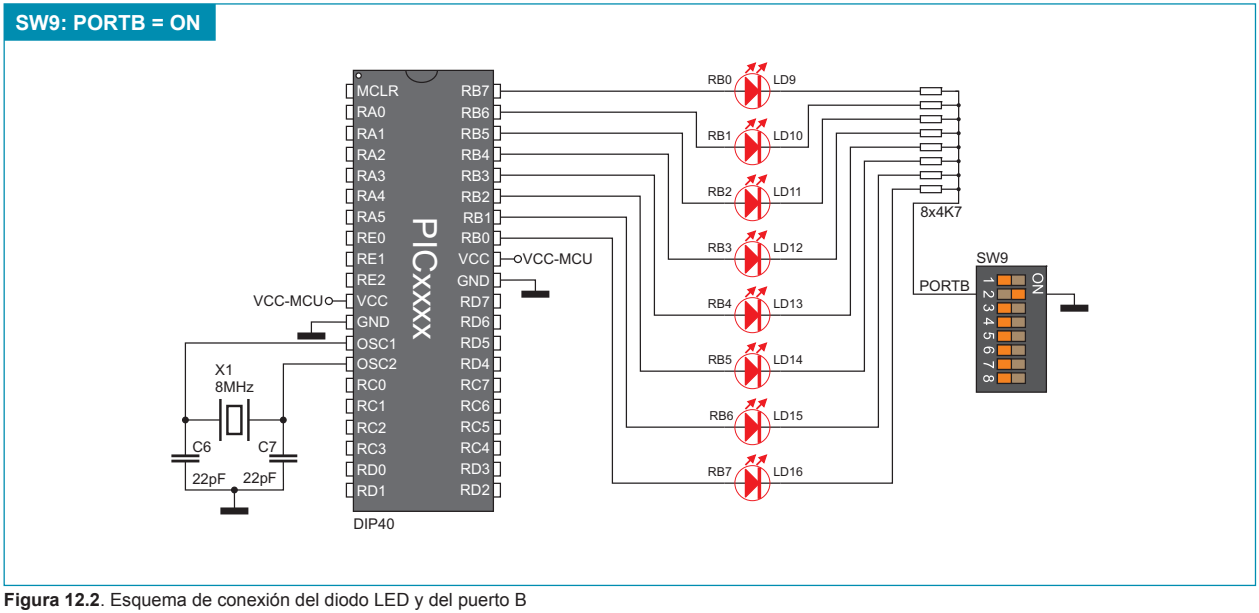


Figura 12.2. Esquema de conexión del diodo LED y del puerto B

13.0. Botones de presión

El estado lógico de todos los pines de entrada digitales del microcontrolador se puede cambiar al utilizar los botones de presión. El puente J17 se utiliza para determinar el estado lógico que será aplicado al pin deseado del microcontrolador al apretar el botón apropiado. El propósito del resistor protector es de limitar la máxima corriente lo que impide la ocurrencia de un corto circuito. Los usuarios com más experiencia pueden, si es necesario, pueden cortocircuitar este resistor utilizando el puente J24. Justamente junto a los botones de presión, se encuentra un botón RESET que no está conectado al pin MCLR. La señal de reset se genera por el programador.

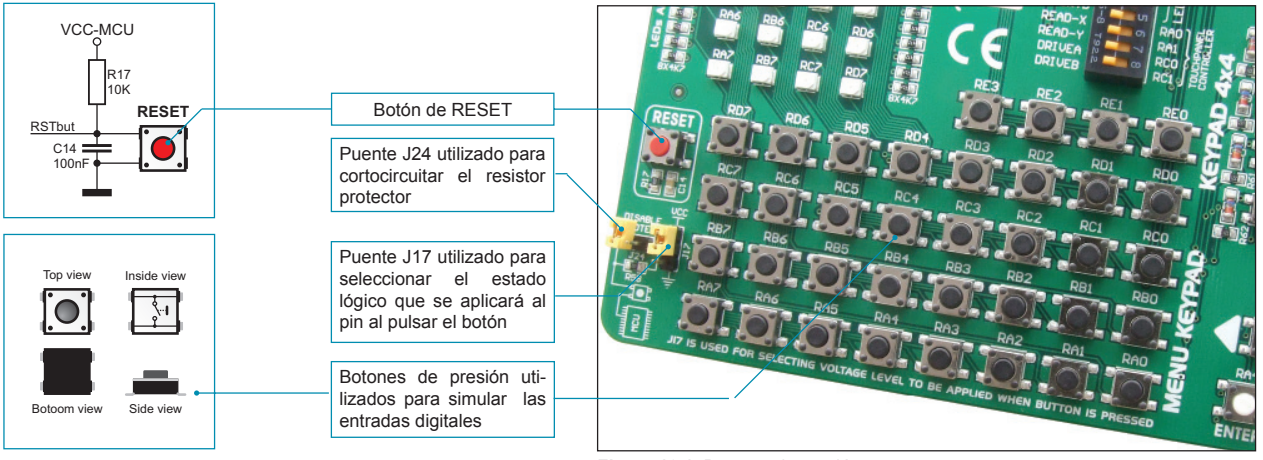


Figura 13.1. Botones de presión

Al pulsar cualquier botón de presión (R0-R7) cuando el puente J17 se encuentre en la posición VCC-MCU, un 1 lógico (5V) será aplicado al pin apropiado del microcontrolador como se muestra en la Figura 13.2.

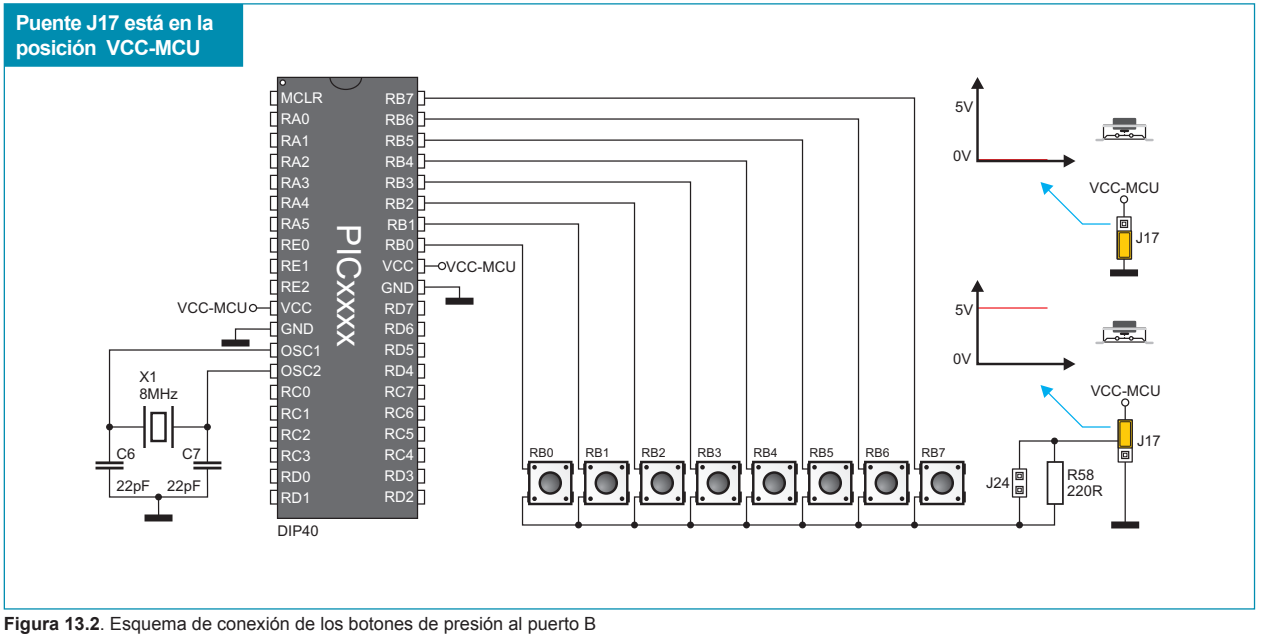


Figura 13.2. Esquema de conexión de los botones de presión al puerto B

14.0. Teclados

En el sistema de desarrollo *EasyPIC6* se encuentran dos teclados. Estos son el teclado 4x4 y el teclado *MENU*.
El teclado 4x4 representa un teclado alfanumérico estándar de 4x4 teclas que está conectado al PORTD del microcontrolador.
El funcionamiento de este teclado está basado en el principio de escanear los pines según el que los pines RD0, RD1, RD2 i RD3 se configuran como las entradas conectadas al resistores *pull-down*. Los pines RD4, RD5, RD6 i RD7 se configuran como las salidas de voltaje da alto nivel. Al presionar cualquier botón un uno lógico se aplicará a los pines de entrada. La detección del botón de presión se raliza por medio de software. Por ejemplo, al presionar el botón de presión “6” un uno lógico (1) aparece en el pin RD2. Para detectar qué botón de presión fue presionado, un uno lógico (1) se aplica a cada pin de entrada: RD4, RD5, RD6 y RD7.
Los botones del teclado *MENU* están conectados de la manera similar a los botones en el puerto PORTA. La única diferencia yace en la disposición de los botones. Los botones del teclado *MENU* están colocados de tal manera que porporcionen una navegación fácil por menús.

Figura 14.1. Teclado 4x4

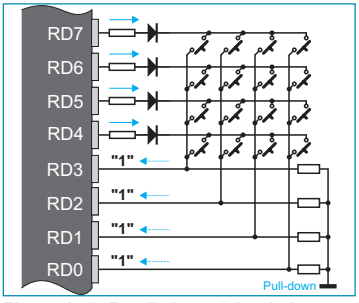


Figura 14.2. Rendimiento del teclado 4x4

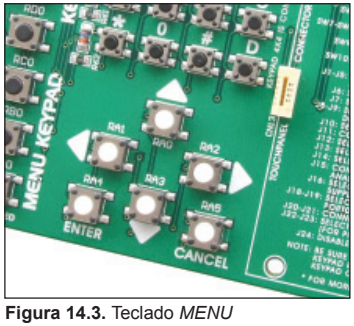


Figura 14.3. Teclado *MENU*

Puente J17 está en la posición VCC-MCU. Los pines RD0, RD1, RD2 y RD3 están conectados a los resistores pull-down por el interruptor DIP SW4.

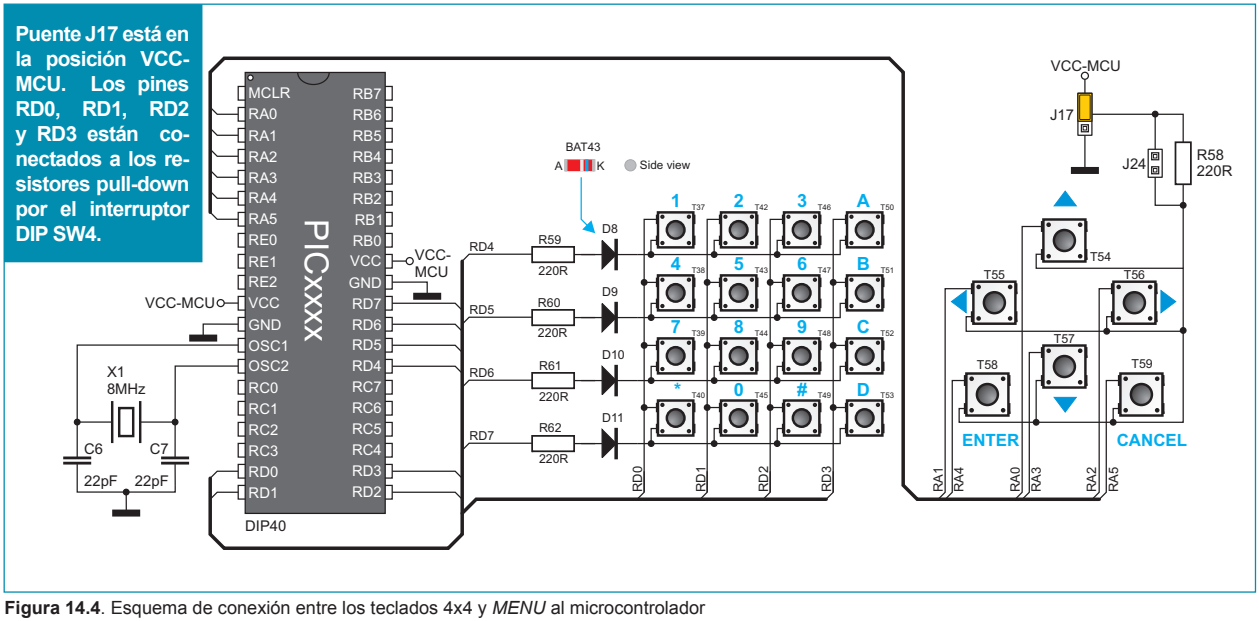


Figura 14.4. Esquema de conexión entre los teclados 4x4 y *MENU* al microcontrolador

15.0. Visualizador LCD 2x16

El sistema de desarrollo *EasyPIC6* dispone de un conector incorporado en el que se coloca el visualizador alfanumérico LCD 2x16. Este conector está conectado al microcontrolador por el puerto PORTB. El potenciómetro P4 se utiliza para ajustar el contraste del visualizador. El interruptor con la etiqueta *LCD - GLCD BACKLIGHT* en el interruptor DIP SW6 se utiliza para encender/apagar luz de fondo del visualizador. La comunicación entre un visualizador LCD y el microcontrolador se establece utilizando el modo de 4 bits. Los dígitos alfanuméricos se visualizan en dos líneas de las que cada una contiene hasta 16 caracteres de 7x5 píxeles.

Figura 15.1. Conector del visualizador alfanúmerico LCD 2x16

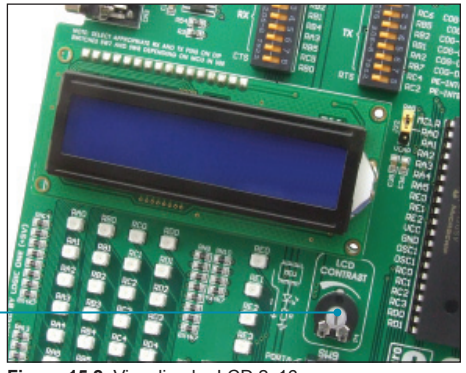


Figura 15.2. Visualizador LCD 2x16

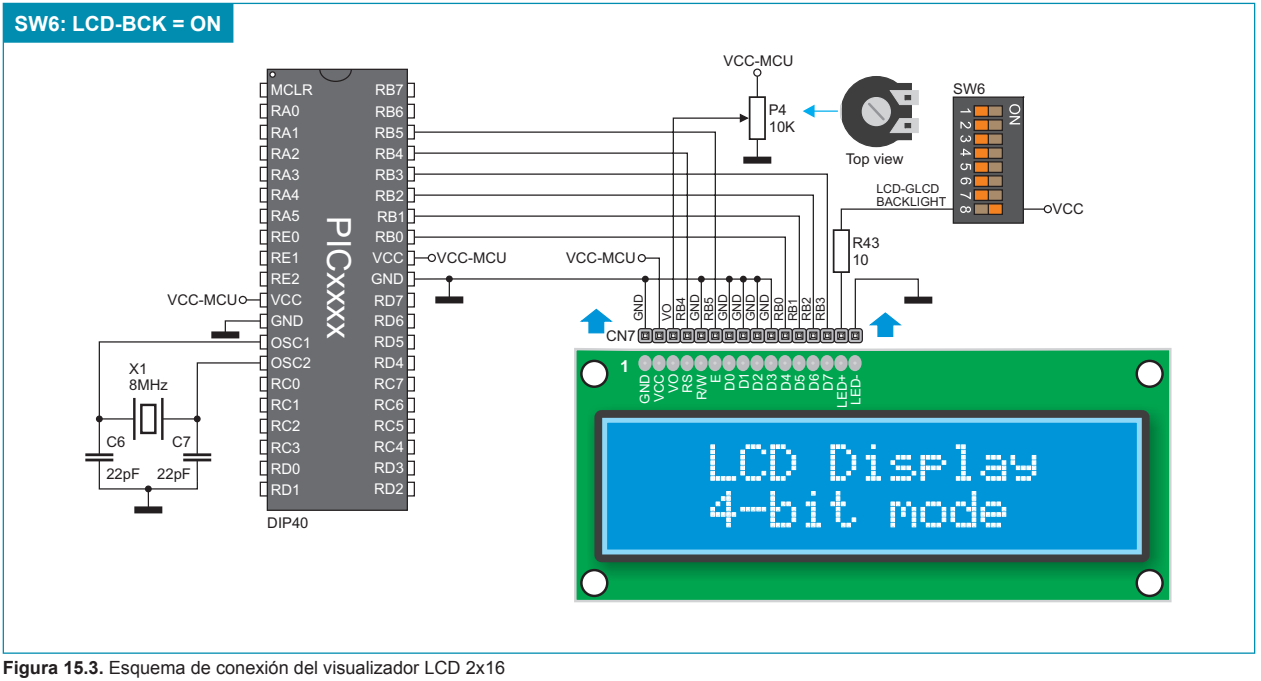
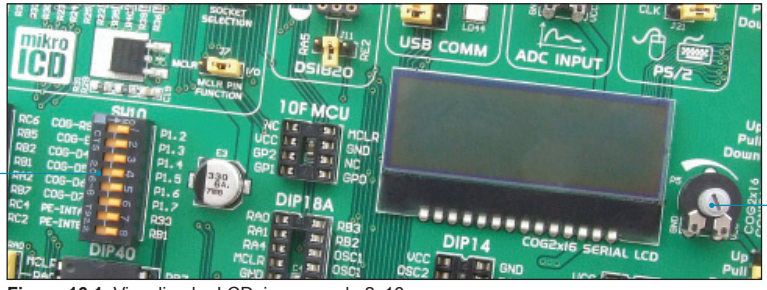


Figura 15.3. Esquema de conexión del visualizador LCD 2x16

16.0. Visualizador LCD incorporado 2x16

El visualizador LCD 2x16 incorporado está conectado al microcontrolador por medio del expansor de puertos. Para utilizar este visualizador es necesario poner todos los interruptores del interruptor DIP SW10 en la posición ON de modo que el visualizador LCD incorporado se conecte al puerto1 del expansor de puertos. El interruptor DIP SW6 permite que el expansor de puertos utilice la comunicación serie. El potenciómetro P5 se utiliza para ajustar el contraste del visualizador.
A diferencia del visualizador LCD, el visualizador incorporado no dispone de la luz de fondo y recibe los datos por medio del expansor de puertos que utiliza la comunicación SPI para comunicarse con el microcontrolador. Este visualizador también visualiza los dígitos en dos líneas de las que cada una contiene hasta 16 caracteres de 7x5 píxeles.

El interruptor DIP SW10 utilizado para enecender el visualizador LCD incorporado 2x16



Potenciómetro para ajustar el contraste

Figura 16.1. Visualizador LCD incorporado 2x16

SW6: CS#, RST, SCK, MISO, MOSI = ON
SW10: 1-8 = ON

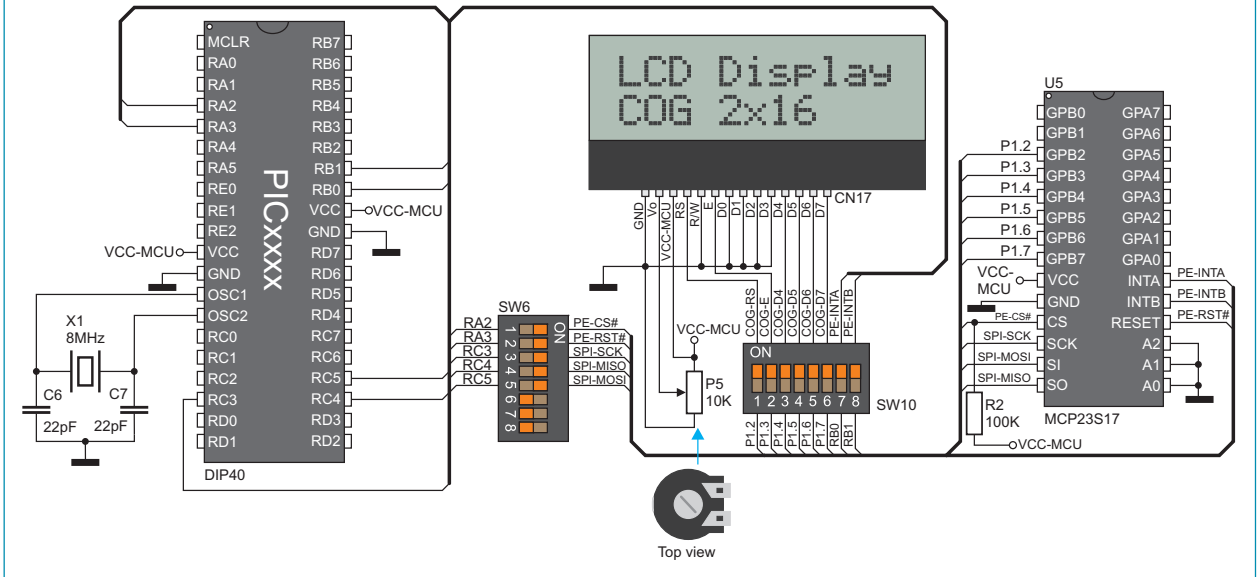


Figura 16.2. Esquema de conexión del visualizador LCD incorporado 2x16

17.0. Visualizador gráfico LCD 128x64

El visualizador gráfico LCD 128x64 (GLCD) proporciona un método avanzado de visualizar los mensajes gráficos. Está conectado al microcontrolador por los puertos PORTB y PORTD. El visualizador GLCD dispone de la resolución de pantalla de 128x64 píxeles que permite visualizar diagramas, tablas u otros contenidos gráficos. Puesto que el puerto PORTB también se utiliza para el visualizador alfanumérico LCD 2x16, no es posible utilizar los dos simultáneamente. El potenciómetro P3 se utiliza para ajustar el contraste del visualizador GLCD. El interruptor 8 en el interruptor DIP SW6 se utiliza para encender/apagar la luz de fondo del visualizador.

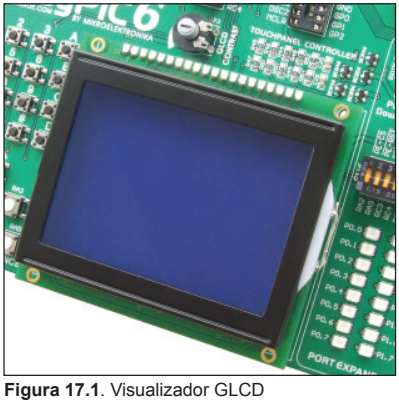


Figura 17.1. Visualizador GLCD

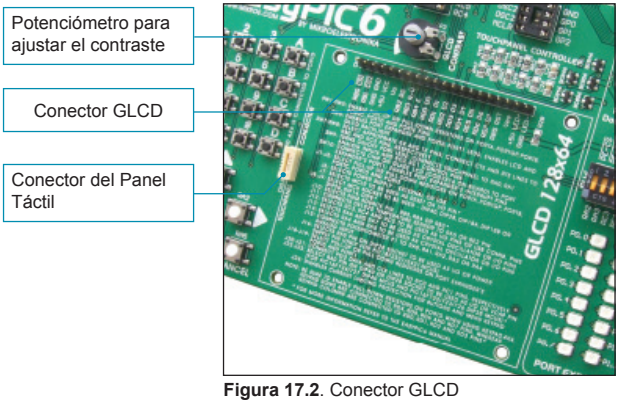


Figura 17.2. Conector GLCD

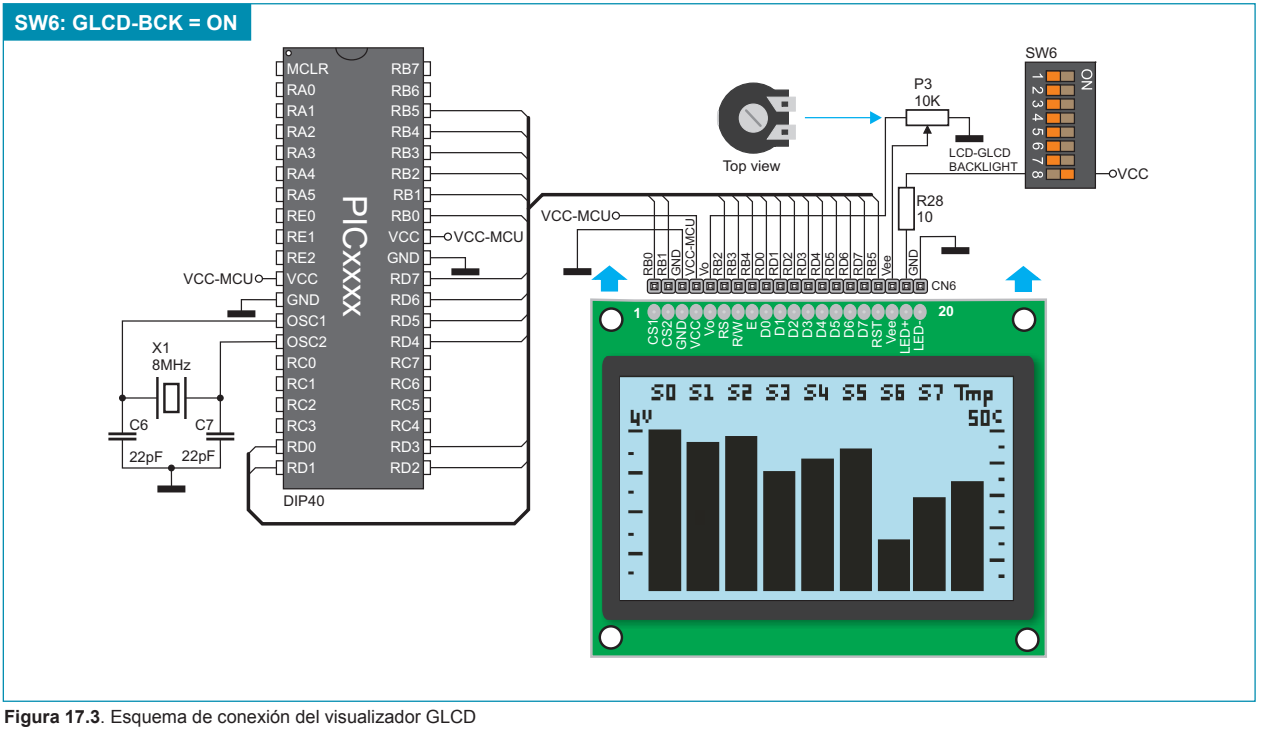


Figura 17.3. Esquema de conexión del visualizador GLCD

18.0. Panel táctil

Un panel táctil es un panel fino, autoadhesivo, transparente y toque senistivo. Se coloca sobre el visualizador GLCD. El propósito principal de este panel es de registrar la presión en un punto específico del visualizador y enviar sus coordenadas en la forma del voltaje analógico al microcontrolador. Los interruptores 5,6,7 y 8 en el interruptor DIP SW9 se utilizan para conectar el panel táctil al microcontrolador.

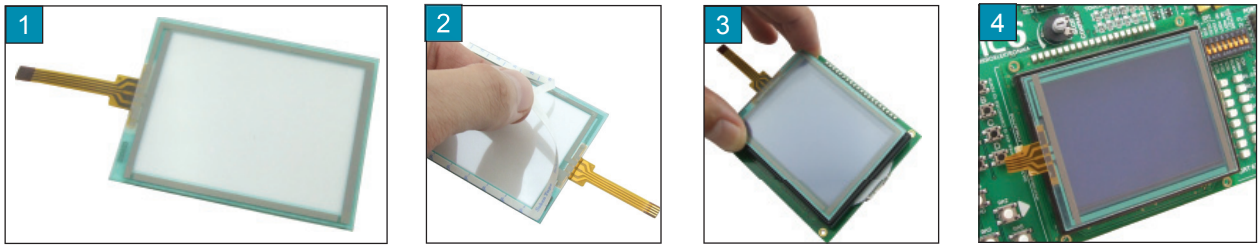


Figura 18.1. Panel táctil

La Figura 18.1. muestra cómo colocar un panel táctil sobre un visualizador GLCD. Asegúrese de que el cable plano esté a la izquierda del visualizador GLCD como se muestra en la Figura 4.

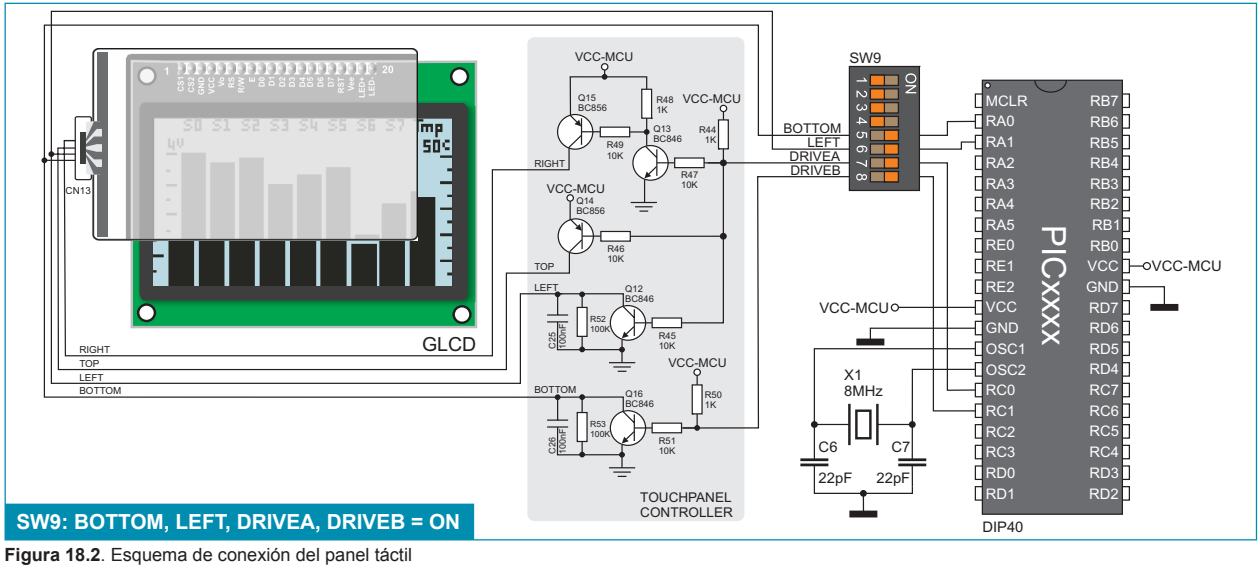


Figura 18.2. Esquema de conexión del panel táctil

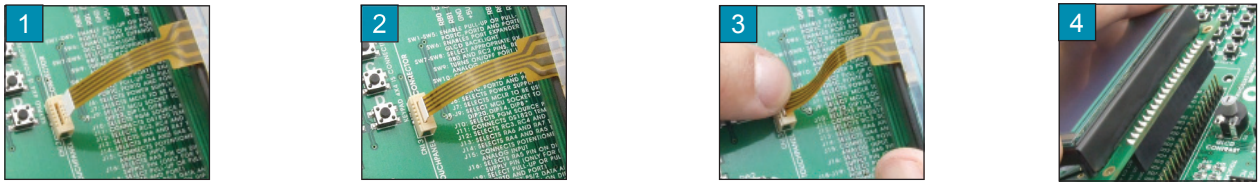


Figura 18.3. Colocación del panel táctil

La Figura 18.3. muestra cómo conectar un panel táctil al microcontrolador. Acerque la punta del cable plano al conector C13 como se muestra en la Figura 1. Inserte el cable en el conector, como se muestra en la Figura 2, y presiónelo lentamente de modo que la punta del cable encaje en el conector completamente, como se muestra en la Figura 3. Luego inserte el visualizador GLCD en el conector apropiado como se muestra en la Figura 4.

NOTA: Los LEDs y los resistores pull-up/pull-down en los pines RA0 y RA1 del puerto PORTA tienen que estar apagados al utilizar un panel táctil.

19.0. Puertos de Entrada/Salida

A lo largo de la parte izquierda del sistema de desarrollo están siete conectores de 10 pines que están conectados a los puertos de Entrada/Salida del microcontrolador. Unos están conectados directamente a los pines del microcontrolador, mientras que otros están conectados por medio de los puentes. Los interruptores DIP SW1-SW5 permiten que cada pin de conector esté conectado a un resistor pull up/pull down. Si los pines de los puertos están conectados a un resistor pull-up o pull-down depende de la posición de los puentes J1-J5.

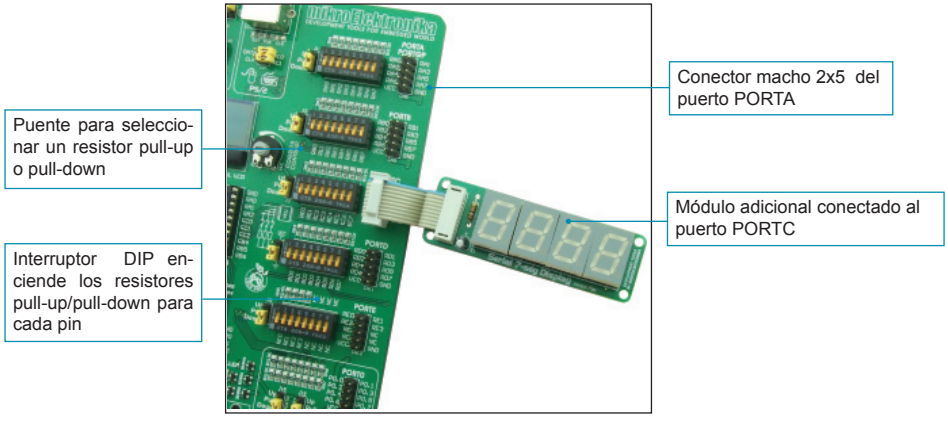

Puente para seleccionar un resistor pull-up o pull-down
Interruptor DIP enciende los resistores pull-up/pull-down para cada pin
Conector macho 2x5 del puerto PORTA
Módulo adicional conectado al puerto PORTC

Figura 19.1. Puertos de E/S

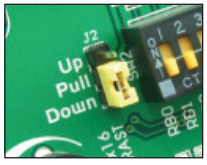


Figura 19.2. Puente J2 en la posición Pull-down

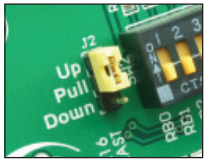


Figura 19.3. Puente J2 en la posición Pull-up

SW2: 1-8 = ON
Puente J2 está en la posición pull-down
Puente J17 está en la posición VCC-MCU

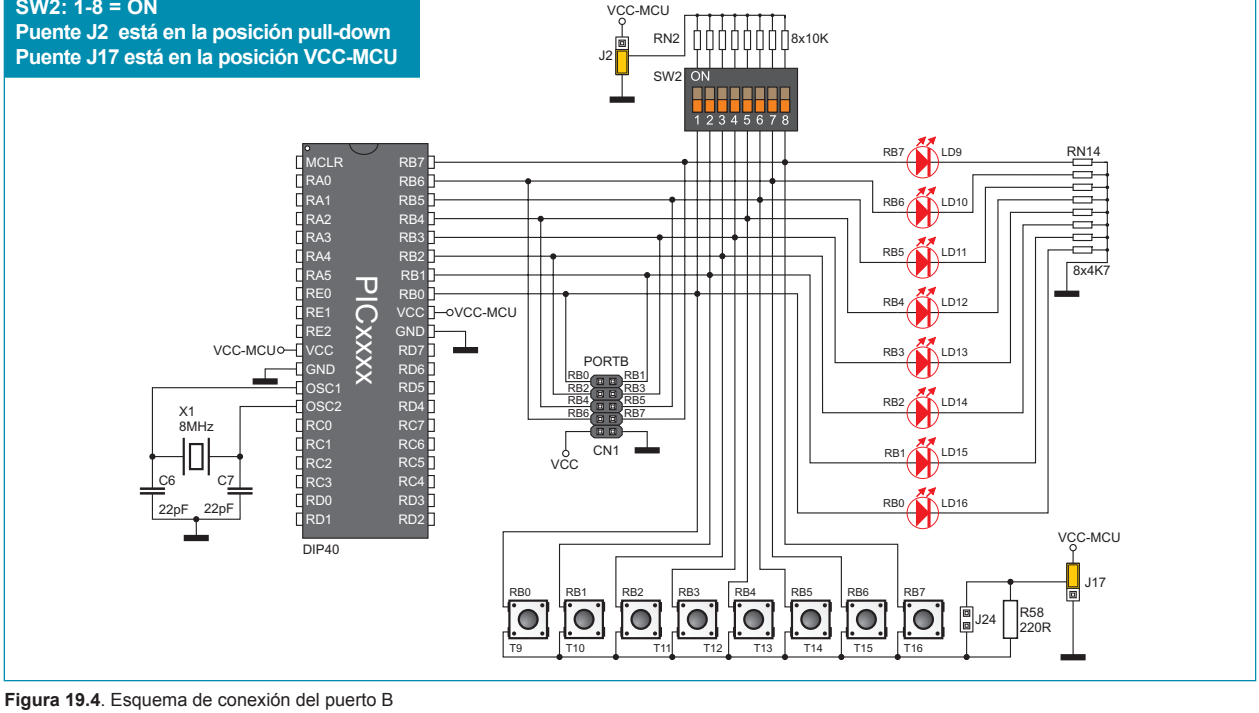


Figura 19.4. Esquema de conexión del puerto B

Los resistores pull-up/pull-down permiten determinar el nivel lógico en todos los pines de entrada del microcontrolador cuando estén en el estado inactivo. El nivel lógico depende de la posición pull-up/pull-down del puente. El pin RB0 junto con el resistor DIP SW2 relevante, el puente J2 y el botón de presión RB2 con el puente J17 se utilizan con el próposito de explicar el funcionamiento de los resistores pull-up/pull-down. El principio de su funcionamiento es idéntico para todos los pines del microcontrolador.

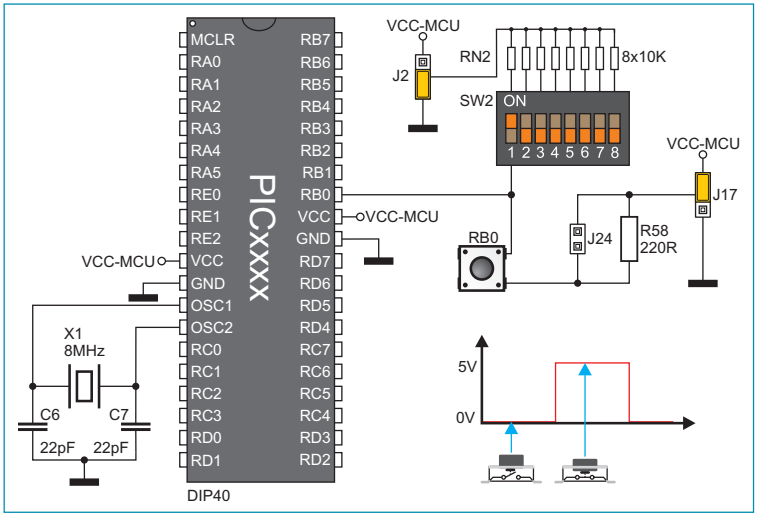


Figura 19.5.Puente J2 en la posición pull-down y el puente J17 en la posición pull-up

Para conectar los pines en el puerto PORTB a los resistores pull-down, es necesario poner el puente J2 en la posición *Down*. Esto permite que se lleve un cero lógico (0) a cualquier pin en el puerto PORTB en el estado inactivo por medio del puente J12 y de la red de resistores de 8x10K. Para llevar esta señal al pin RB0, es necesario poner el interruptor RB0 en el interruptor DIP SW2 en la posición ON.
Por consiguiente, cada vez que se presiona el botón de presión RB0 un uno lógico (1) aparecerá en el pin RB0, con tal de que el puente J17 esté colocado en la posición VCC.

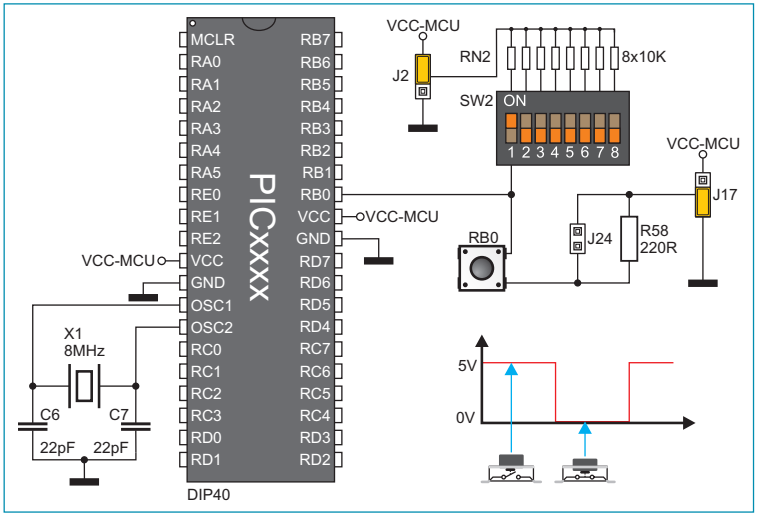


Figura 19.6. Puente J2 en la posición pull-up y puente J17 en la posición pull-down

Para conectar los pines en el puerto PORTB a los resistores pull-up, y para llevar un cero lógico (0) a los pines de entrada en el puerto, es necesario poner el puente J2 en la posición *Up* (5V) y el puente J17 en la posición GND (0V). Además, el pin RB0 en el interruptor DIP SW2 se debe colocar en la posición ON. Esto permite llevar un uno lógico (1) a todos los pines de entrada del puerto PORTB en el estado inactivo por medio del resistor de 10k. El voltaje se lleva al pin RB0 por medio del resistor de 10k y el interruptor RB0.
Por consiguiente, cada vez que se presiona el botón de presión RB0 un cero lógico (0) aparecerá en el pin RB0.

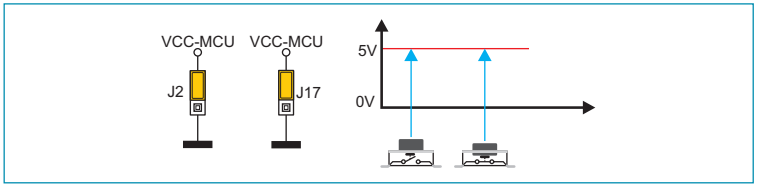


Figura 19.7. Puentes J2 y J17 en las mismas posiciones

En caso de que los puentes J2 y J17 tengan el mismo estado lógico, al presionar cualquier botón de presión no cambia el estado lógico de los pines de entrada.

20.0. Expansor de puertos

Las líneas de comunicación SPI y el circuito MCP23S1 proporcionan el sistema de desarrollo *EasyPIC6* con recursos de incrementar en dos el número de los puertos de E/S. Si el expansor de puertos está conectado al microcontrolador por medio del interruptor SW6, los siguientes pines RA2, RA3, RC3, RC4 y RC5 se utilizarán para la comunicación SPI, por lo que no se pueden utilizar como los pines de E/S. Los interruptores INTA e INTB en el interruptor DIP SW10 habilitan una interrupción. MCP23S17 habilita la expansión paralela de 16 bits y se puede configurar para funcionar en el modo de 16 o 8 bits.

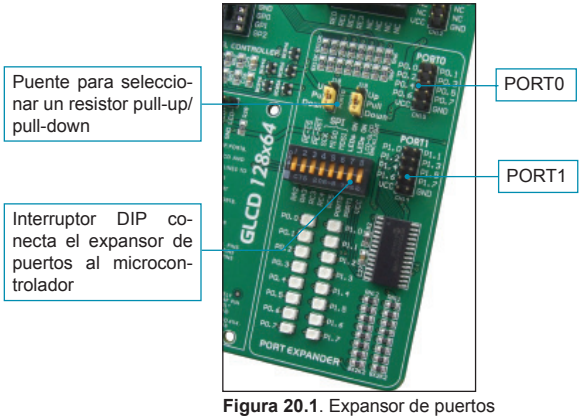


Figura 20.1. Expansor de puertos

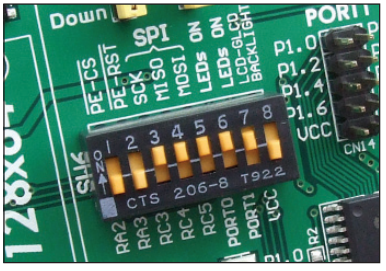


Figura 20.2. Posición del interruptor DIP SW6 cuando el expnsor de puertos está habilitado

SW6: CS#, RST, SCK, MISO, MOSI = ON
SW10: INTA, INTB = ON
Puentes J18 y J19 en la posición superior

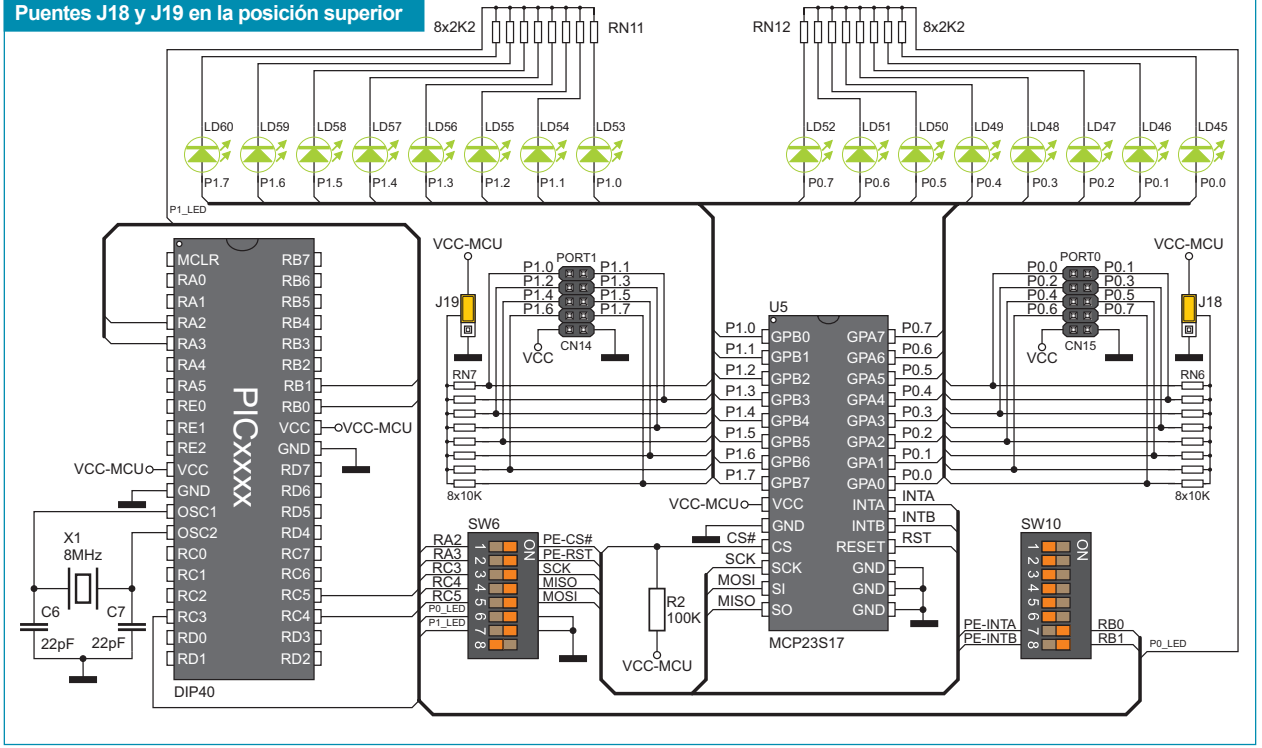


Figura 20.3. Esquema del expansor de puertos

TÉRMINOS Y CONDICIONES

Todos los productos de MikroElektronika son protegidos por la ley y por los tratados internacionales de derechos de autor. Este manual es protegido por los tratados de derechos de autor, también. Es prohibido copiar este manual, en parte o en conjunto sin la autorización previa por escrito de MikroElektronika. Se permite imprimir este manual en el formato PDF para el uso privado. La distribución y la modificación de su contenido son prohibidas.

MikroElektronika proporciona este manual “como está” sin garantías de ninguna especie, sean expresas o implícitas, incluyendo las garantías o condiciones implícitas de comerciabilidad y aptitud para fines específicos.

Aunque MikroElektronika ha puesto el máximo empeño en asegurar la exactitud de la información incluida en este manual, no asume la responsabilidad de ninguna especie de daños derivados del acceso a la información o de los programas y productos presentados en este manual (incluyendo daños por la pérdida de los beneficios empresariales, información comercial, interrupción de negocio o cualquier otra pérdida pecuniaria).Las informaciones contenidas en este manual son para el uso interno. Pueden ser modificadas en cualquier momento y sin aviso previo.

ACTIVIDADES DE ALTO RIESGO

Los productos de MikroElektronika no son tolerantes a fallos y no están diseñados, fabricados o pensados para su uso o reventa como equipo de control en línea en entornos peligrosos que requieran un funcionamiento sin fallos, como en instalaciones nucleares, en la navegación aérea o en sistemas de comunicaciones, de tráfico aéreo, máquinas de auxilio vital o sistemas de armamento, en los que un fallo del software podría conducir directamente a la muerte, lesiones corporales o daños físicos o medioambientales graves (“Actividades de alto riesgo”). MikroElektronika y sus proveedores niegan específicamente cualquier garantía expresa o implícita de aptitud para Actividades de alto riesgo.

MARCAS REGISTRADAS

Los productos y los nombres corporativos utilizados en este manual son protegidos por la ley de los derechos de autor, sin reparar en la ausencia de notas adicionales. Las marcas registradas son utilizadas exlusivamente con el propósito de identificar y explicar los conceptos correspondientes y en beneficio de sus respectivos propietarios, sin intención de infringirlas.

Copyright© 2003 – 2009 por MikroElektronika. Todos los derechos reservados.

Si quiere saber más de nuestros productos, por favor visite nuestra página web www.mikroe.com

Si tiene problemas con cualquiera de nuestros productos o sólo necesita información adicional, deje un ticket en www.mikroe.com/en/support

Si tiene alguna pregunta, comentario o propuesta de negocio, póngase en contacto con nosotros en office@mikroe.com